Jurassic News

Retrocomputer Magazine

Anno 3 - Numero 19 - Ottobre 2008



Jurassic News

Rivista aperiodica di Retro-computing

Coordinatore editoriale
Salvatore Macomer [Sm]

Redazione
Sonicher [Sn]
redazione@jurassicnews.com

Hanno collaborato a questo numero:
Tullio Nicolussi [Tn]
Lorenzo 2 [L2]
Besdelsec [Bs]
Mister X [Mx]
Bob Lash

Impaginazione e grafica
Anna [An]

Diffusione
marketing@jurassicnews.com

La rivista viene diffusa in formato PDF via Internet agli utenti registrati sul sito www.jurassicnews.com.
la registrazione è gratuita e anonima; si gradisce comunque una registrazione nominativa.

Contatti
info@jurassicnews.com

Jurassic News promuove la libera circolazione delle idee

Ottobre 2008

In Copertina

Come un Sole nascente dalle tenebre dell'era priva di calcolatori per uso personale, il SOL della Processor Technology squarcia il telo del tempio e si erge ad indicare la strada, quella strada che porterà diritto al cellulare che ora tenete in mano...

Editoriale

Pubblicare, che passione!

Sono reduce da un lungo scambio di e-mail con degli amici, conosciuti sul Web, che avrebbero voluto dare vita ad una rivista simile alla nostra ma dedicata esclusivamente al retro gaming.
Iniziativa lodevole, da me incoraggiata e sostenuta con dei consigli che ritenevo doveroso elargire sia perché mi erano stati esplicitamente chiesti, sia perché so quali sono le trappole di una iniziativa simile e volevo, per quanto in mio potere, aiutare questo gruppo di amici.
Il problema è che quando si parte con un obiettivo del genere, si è colmi di entusiasmo e di idee, ma l'entusiasmo è messo a dura prova fin dalle prime difficoltà e le idee, che sembravano tanto buone prima di partire, finiscono spesso per essere di intralcio invece che di stimolo. Scrivere poi non è da tutti, e non lo dico per falsa modestia. Moltissime persone sanno benissimo quello che vorrebbero dire ma non riescono a farlo. Parlare è una cosa, scrivere un'altra!
Il primo consiglio è quello di partire con i piedi di piombo: concentratevi su poche rubriche e poche pagine e non pensate di coinvolgere editori per farne una versione cartacea, se volete evitarvi una cocente delusione.
L'altro consiglio fondamentale è quello di partire solo quando avete abbastanza materiale per fare tre numeri completi della rivista e non azzardare una periodicità mensile, ma darsi il tempo necessario.
Gli amici di cui ho parlato hanno poi rinunciato, ne ho seguito l'iter nel forum che avevano attivato. Una decina le adesioni: "sì sì che bello, io scrivo di questo e di quello,..." poi qualcuno sollecita gli articolisti ad inviare il materiale promesso: nulla, evidentemente impallati davanti alla pagina bianca, niente, non rispondono nemmeno se non con qualche scusa... la nonna malata, la zia con il morbillo, la suocera in visita... Buca anche questa.
Ne ho voluto parlare per farvi apprezzare ancora di più la nostra umile opera.

[Sm]

Jurassic News

è una fanzine dedicata al retro-computing nella più ampia accezione del termine. Gli articoli trattano in generale dell'informatica a partire dai primi anni '80 e si spingono fino ...all'altro ieri.

La pubblicazione ha carattere puramente amatoriale e didattico, tutte le informazioni sono tratte da materiale originale dell'epoca o raccolte (e attentamente vagliate) da Internet.

Normalmente il mate-riale originale, anche se "jurassico" in termini informatici, non è privo di restrizioni di utilizzo, pertanto non sempre è possibile riportare per intero articoli, foto, schemi, listati, etc…, che non siano esplicitamente liberi da diritti.

La redazione e gli autori degli articoli non si assumono nessuna responsabilità in merito alla correttezza delle informazioni riportate o nei confronti di eventuali danni derivanti dall'applicazione di quanto appreso sulla rivista.

Retrocomputing

La forza del destino

Prima o poi succede: il computer cui siamo tanto affezionati non si accende più.

E' proprio così, che ci piaccia o no: le macchine si rompono. Si rompono anche gli umani, figuriamoci le macchine!

L'inevitabile accade all'improvviso, principalmente quando si accende un sistema lasciato "a dormire" per troppi mesi, si scopre che semplicemente non parte. Ci consola la circostanza che almeno ci è stato risparmiato il tragico momento del trapasso, il sistema è passato a "miglior vita" durante il sonno: una morte dolce per un umano, sicuramente da sottoscrivere in pieno, interlocutoria per l'amato home: quando questo è accaduto? Non sarà stato quel colpetto dato involontariamente durante lo spostamento di altri fratelli elettronici ivi depositati?

Dato che abbiamo poche armi per opporci a questo ineluttabile accadimento vediamo almeno quali azioni possiamo mettere in campo prima (meglio) e dopo il trapasso.

Da un lato il desiderio di ogni appassionato di retro computer è quello di mantenere funzionante la propria collezione a tempo indefinito, dall'altro c'è un desiderio quasi fisico di recuperare alla vita vecchi sistemi trovati nei cassonetti o comprati/scambiati con altri appassionati come parti di ricambio.

La prima idea è proprio quella della ricambistica, come si definisce l'insieme delle tecniche che tendono a rendere disponibili i pezzi di ricambio per la sostituzione di quelli guasti.

Immagazzinare un certo (meglio se cospicuo) numero di elementi da usare come "spare part" è ovviamente una tecnica diffusa e da seguire per quanto possibile.

Del resto chi di noi non si porterebbe a casa qualsiasi numero di macchine, anche scassatissime, se le trovasse gratis e soprattutto se fossero dello stesso tipo del "pezzo più prezioso della nostra collezione" ? (Tutti hanno il pezzo più prezioso!). Il limite, oltre alla decenza, è rappresentato dallo spazio di cui si dispone, dal momento che pochi di noi possono permettersi di affittare un capannone allo scopo.

Un comportamento conservativo corretto non può basarsi solamente sulla disponibilità di parti di ricambio "ad libidum". Semplicemente non sarebbe una strategia vincente nel lungo periodo. Va considerato infatti che solo parzialmente il degrado delle parti componenti è legato all'uso. Può essere vero per i

pezzi contenenti parti in movimento, che presumibilmente si consumano nella rotazione, mentre per i circuiti elettronici in genere esiste anche un "orologio biologico" inarrestabile che porta ogni singolo chip o anche ogni singolo stampato verso la vecchiaia.

Questo significa che anche le piastre di ricambio e i drive per floppy immagazzinati per riserva, sono soggetti ad invecchiamento.

La strategia di conservazione ha come gol finale il rendere disponibile un sistema funzionante nel corso del tempo. Da questo semplice corollario deve seguire l'implementazione di quelle azioni che possono assicurare il raggiungimento dell'obiettivo massimizzando il risultato pur tenendo conto dei criteri di economicità e dei vincoli che inevitabilmente sono presenti.

Diamo per scontato che tutti conoscano e seguano i corretti comportamenti "elementari" che riguardano il corretto immagazzinamento (no ambienti umidi, polverosi, etc...). Parliamo invece delle strategie "più raffinate" e per forza di cose anche molto più dispendiose in termini di tempo.

Prima di tutto voglio fare un'affermazione che può apparire elementare, ma per non lasciare zone d'ombra va detta e che è la seguente: un sistema di calcolo funzionante non serve a nulla se non è correlato del necessario software di base. Ad esempio un sistema CP/M senza floppy di boot serve francamente a poco! Sono anni che cerco un floppy per bootare un sistema Questar/M della Honeywell ma senza successo!

Quindi la conservazione riguarda in egual misura hardware e software. Della conservazione del software abbiamo già trattato qui e altrove.

Il decalogo del "buon conservatore" parte dalla selezione dei sistemi. E' un argomento spinoso da affrontare, siamo perfettamente d'accordo, infatti tutti noi vorrebbero accrescere sempre di più la propria collezione, non limitarla. La numerosità fa a pugni con la qualità della conservazione, che può essere portata a compimento (sempre nell'ottica del lungo periodo), solo concentrandosi su un insieme limitato di sistemi, in proporzione diretta con il tempo/risorse di cui disponiamo.

Una volta deciso dove rivolgere i propri sforzi scatta il piano di conservazione che deve considerare tutti gli aspetti della questione.

Due sistemi uguali e funzionanti è un'ottima misura di sicurezza. Questa duplicazione ci consente infatti da una parte di avere un ricambio "certo" e dall'altra di disporre di un sistema di "confronto" da interrogare selettivamente. Una delle tecniche più utilizzate, anche perché elementare, dai riparatori, è quella di disporre di una apparecchiatura sicuramente funzionante sulla quale testare uno

ad uno le parti "sospette" dell'apparecchio defunto.

Un'altro fattore che rende la riparazione del guasto un progetto di buona riuscita è l'esperienza. Beh, voi direte, quella bisogna farsela, mica si compra o si eredita per grazia ricevuta! E' sicuramente così, anche se esiste un "trucco" per accellerare l'acquisizione di capacità di analisi e diagnostica. E' un trucco che veniva usato dalle vecchie generazioni di artigiani, i quali prima mettevano il "giovane di bottega" a guardare e poi gli permettevano di "farsi le ossa" montando e smontando a tempo perso certe parti di macchine. Molto spesso questa "esperienza" deriva da una certa familiarità con il soggetto sottoposto a riparazione e in generale da una manualità che rende le mosse del riparatore sicure e decise. Quando non è necessario arrovellarsi il cervello per capire come smontare quella certa cosa, allora è più facile dedicarsi serenamente al core del problema, cioè il guasto, senza il timore di rompere qualcosa nel maneggiare l'apparecchiatura.

Schemi e manuali devono fare parte del corredo dell'hobbista che si dedica al mantenimento e riparazione (si può affermare che i due termini sono sinonimi). Per nostra fortuna gli schemi elettrici dei primi home sono elementari: in una paginetta ci stà tutto! In queste condizioni è facile seguire il segnale da controllare sulle piste e ricavare l'ubicazione dei punti da misurare.

Le misure che possiamo fare su un circuito digitale a microprocessore sono di due tipi: statiche e dinamiche. Assumiamo che nessuno disponga di sofisticati probe digitali, che costano parecchie migliaia di Euro, ma che nel nostro piccolo laboratorio trovi posto il classico tester, un oscilloscopio e al limite una sonda logica.

Le prove statiche possono limitarsi alla verifica della corretta alimentazione dei chip e della continuità delle piste (critici in questo senso sono i connettori). Le prove dinamiche più elementari consentono di verificare piccole slice di tempo che dovrebbero fotografare un comportamento noto del chip sotto esame. Ad esempio quando si resetta un processore si può sicuramente seguire la sequenza di segnali emessi dalla procedura di reset e che si riflettono sui circuiti esterni.

Lasciamo questi argomenti molto tecnici per tornare a indicazioni generali e al tema che ci impegna: la conservazione in buono stato dei nostri sistemi di calcolo ormai obsoleti.

Può accadere che un particolare componente non sia più reperibile. Sono passati pochi anni ma, come sa bene chi ci si è provato a costruire "repliche", può essere difficile trovare addirittura un pulsante di certe dimensioni, per non parlare di Prom e in generale di circuiti custom.

La customizzazione dei circuiti, avvenuta praticamente da subito: chi non ricorda il chip "Ferranti" nello ZX81 della Sinclair? E per forza:

è stata proprio questa possibilità a far scendere drasticamente complessità circuitale e costi di produzione!

Ora se perdete una Prom è praticamente impossibile reperirne una uguale, sempre se non disponete delle famose spare parts così preventivamente immagazzinate.

Uguale discorso per le Eprom e addirittura per le RAM i cui modelli "prima maniera" non sono più prodotti. Che fare in questo caso? E' necessario rimpiazzare il pezzo originale con un circuito "facente funzioni", cosa tutt'altro che semplice!

Per fortuna ci sono gruppi di appassionati che per le piattaforme più diffuse ed amate, ha trovato da tempo la risposta e in qualche caso fornisce addirittura circuiti pronti per sostituire quelli custom originali.

In ogni caso non sarebbe male scaricare per tempo il contenuto binario delle ROM e conservarlo per ogni eventualità. Qui ci potrebbe essere di enorme aiuto un programmatore di Eprom/Prom, a patto che supporti i vecchi chip. nel caso delle ROM di sistema non è affatto difficile ottenerne un dump direttamente sulla piattaforma sulla quale sono installate. In fondo si tratta sempre di locazioni di memoria indirizzabili se non da BASIC almeno dall'Assembler. Il difficile semmai è trasferire il file binario (o suo equivalente ASCII) su PC per registrarlo su supporti sicuri.

Nella conservazione dei sistemi l'aspetto di "interoperabilità" delle piattaforme è essenziale. Infatti tutti i professionisti in questo campo (ci sono praticamente solo negli States), consigliano caldamente di realizzare il collegamento fra un sistema e un equivalente moderno al fine di permettere questo viaggio bi-direzionale delle informazioni di sistema, oltre che del software necessario al funzionamento della piattaforma.

Conclusione.

La forza del destino impone una legge spietata: prima o poi i sistemi si rompono e prima o poi la gente che è in grado di ripararli sparisce anch'essa. I progetti di conservazione dei calcolatori impone un rigido protocollo di azioni che rendano la dipartita delle nostre macchine un evento controllabile nel limite del possibile.

L'oblio dei sistemi è poi evitato da una attenta raccolta e messa a disposizione di tutte quelle informazioni che consentano ai posteri un efficacie intervento di recupero o, al limite, di ricostruzione.

[Tn]

Retro Riviste

La rassegna dell'editoria specializzata dai primi anni '80 ad oggi

Scheda

Titolo:
Commodore
Sottotitolo:
Per utenti VIC-C64-C16-Plus4 e C128

Editore:
System Eitoriale
Lingua:
Italiano
Prezzo:
L 3.000
Pagine:
60

Primo numero:
Luglio 1984
Ultimo numero:
Dicembre 1985
Poi:
Personal Computer

COMMODORE



La rivista COMMODORE rivela fin dal nome la sua mission: i computer della Commodore: VIC, C64, 16, Plus-4 e C128. Il primo numero esce nel luglio 1984 e l'editore System Editoriale riesce a mantenere il periodico per un anno e mezzo fino al dicembre 1985 quando con il numero 15 viene annunciata la fusione di tre periodici in un'unica iniziativa trasversale. Per essere precisi dovrebbe esistere anche un numero 16 del gennaio 1986 ma non ne ho trovata la prova certa.

In un mercato molto in fermento ma che cominciava già a mostrare qualche abbandono da parte degli editori, le riviste Commodore, Sinclair Computer e MSX vanno a fondersi nell'unica testata chiamata Personal Computer. Evidentemente la polverizzazione eccessiva delle marche/modelli ha nociuto all'editoria che stentava a trovare una sua diffusione mumerica soddisfaciente.

Della rivista Commodore si hanno successivamente a questa fusione due strascichi monografici: il primo dedicato al linguaggio Assembly e alla grafica e il secondo al Drive floppy.

La System Editoriale fornisce un raro esempio di apertura mentale, almeno per l'Italia: permette infatti la pubblicazione di alcuni suoi periodici on-line in versione scansionata. Pagine della rivista Commodore si possono trovare sul sito di Amici di Hal (http://www.museo-computer.it/SalaLettura/Commodore) e su Ready64 (http://ready64.org/download/categorie.php?cat=27) dove invece si trovano le immagini di alcuni dei floppy allegati alla rivista stessa.

Si tratta di una rivista mono-marca e come tale orientata al software specifico della piattaforma che va a coprire. L'approfondimento dei temi è vario, per coprire le necessità del principiante e dell'uti-

lizzatore maggiormente scafato.

La qualità dei contenuti cresce di numero in numero, tanto che si arriva al numero di chiusura (il 15^) dove si apprende con una certa sorpresa la fine della rivista, anche se essa va a formare l'iniziativa congiunta con le altre due piattaforme Sinclair e MSX di cui si è detto.

Ci chiediamo il senso di questa scelta, in una realtà dove l'utilizzatore non possedeva molteplici piattaforme (il computer costava lo stipendio da tre mesi a un anno a secondo della sofisticazione e delle periferiche che si volevano collegare).

Da una intervista a Michele di Pisa, pubblicata sul sito di Ready64 si evince la storia di questa testata racchiusa in poche righe:

"Il successo improvviso e inaspettato di Commodore Computer Club mi fece subito temere che gli altri editori non sarebbero rimasti inerti a guardare. Temevo che qualcuno dei grandi lanciasse a breve scadenza una rivista concorrente. Forse uno dei massimi ci ha anche provato. Memore di quanto avevano fatto Gianfranco Binari e Daniel Caimi, con Suono e Stereoplay, pensai che la prevenzione migliore sarebbe stata quella di tentare di saturare il mercato diventando io stesso il mio principale concorrente. Fu così che convinsi Gloriano Rossi, uno dei migliori conoscitori dei computer Commodore, ad assumere la direzione della nuova pubblicazione. Gloriano lo fece col massimo impegno e per tutto il primo anno, la rivista registrò un venduto medio di 20-22.000 copie".

Personalmente posseggo il numero speciale dedicato alla grafica e all'assembler. In qualche modo si tratta di articoli che trattano cose abbastanza elementari e conosciute ai più, ma ciononostante è interessante vedere come l'editore abbia voluto rendere omaggio ad una piattaforma e alla testata stessa pubblicandone gli aspetti più particolari in un'unico numero monografico.

[Sn]









Retro Linguaggi

La storia dell'informatica è stata anche la storia dei linguaggi di programmazione.

ABAP (parte 7)



Una delle strutture fra le più importanti del linguaggio ABAP è il concetto di "tabella interna". Una tabella interna è esattamente come una tabella definita nel DB, con la differenza che essa vive in memoria: viene definita come struttura dati all'interno di un programma, caricata ed elaborata con dati ed infine cancellata al termine dell'utilizzo.

Listato 1.

```
REPORT ZMX_07_01                    .

TYPES: BEGIN OF RECORD,
        NOME(25)    TYPE C,
        COGNOME(25) TYPE C,
      END OF RECORD.

DATA WA   TYPE RECORD.
DATA ITAB TYPE STANDARD TABLE OF RECORD.

START-OF-SELECTION.

  CLEAR WA.
  MOVE 'MARIO' TO WA-NOME.
  MOVE 'ROSSI' TO WA-COGNOME.
  APPEND WA TO ITAB.

  CLEAR WA.
  MOVE 'LUIGI'  TO WA-NOME.
  MOVE 'VERDI'  TO WA-COGNOME.
  APPEND WA TO ITAB.

  LOOP AT ITAB INTO WA.
   WRITE: WA-NOME, WA-COGNOME.
   NEW-LINE.
  ENDLOOP.

END-OF-SELECTION.
```

Vediamo come si definisce, cominciando con i concetti più semplici.

Prima di tutto va definito una sorta di "record" che formerà la tabella stessa. Ad esempio definiamo un tipo di dato che chiamiamo proprio "RECORD", ma qualunque nome sarebbe valido comunque.

```
TYPES: BEGIN OF RECORD,
        NOME(25)    TYPE C,
        COGNOME(25) TYPE C,
      END OF RECORD.
```

La definizione di un "tipo utente" con la direttiva "TYPES", non alloca memoria e quindi il nostro nome "RECORD" non è direttamente utilizzabile nel programma. Va definita una variabile di quel tipo:

```
DATA WA TYPE RECORD.
```

La nostra variabile di lavoro si chiamerà WA (che starebbe per Working Area, una usanza molto comune nei sorgenti ABAP).

Infine la dichiarazione della tabella interna che abbiamo chiamato ITAB (ma la potevamo chiamare "pippo", come si sà...).

```
DATA ITAB
  TYPE STANDARD TABLE
  OF RECORD.
```

Nel sorgente visibile nel Listato 1, abbiamo effettuato due caricamenti di record, prima preparando la struttura di appoggio WA e poi con "APPEND", inserendo il record in coda a quelli esistenti nella tabella ITAB.

Al termine del caricamento abbiamo innescato un LOOP di utilizzo della tabella stessa. Questa istruzione è simile alla SELECT fatta su una tabella del DB: elenca i record che fra la LOOP e la ENDLOOP sono disponibili all'elaborazione (una semplice stampa, ad esempio, come quella visibile in figura 1).

L'accoppiata: tabella interna + tabelle nel DB, è un punto di forza incredibile dell'ambiente SAP, che rende produttivo al massimo lo sviluppo di applicativi. Un utilizzo molto frequente è quello del caricamento di una tabella interna con dati provenienti da una o più tabella del DB, l'elaborazione in memoria ed eventualmente la riscrittura dei dati nel DB stesso. Va rilevato che SAP non ha problemi di capacità di memoria e quindi le tabelle interne possono tranquillamente contenere milioni di record, ovviamente bisogna attrezzare opportunamente il server.

Nel listato 2 vediamo un utilizzo misto di dati provenienti da una tabella del DB, la ben conosciuta SCARR, che vengono immagazzinati nella tabella interna ITAB.

La ITAB è stata dichiarata proprio di tipo "come la SCARR", utilizzando la direttiva "LIKE".

Abbiamo poi un loop di caricamento dei dati che attraverso la SELECT vengono letti dalla SCARR e trasferiti pari-pari nella struttura di appoggio WA ed infine "appesi" nella ITAB.

Al termine del caricamento abbiamo innescato un LOOP/ENDLOOP che utilizza i dati per l'elaborazione (una semplice WRITE, ma non importa la complessità dell'elaborazione stessa in questo esempio).

Non mostriamo il risultato dell'elaborazione, banalmente ricavabile

Figura 1.

Listato 2.

```
REPORT  ZMX_07_02                    .

TABLES: SCARR.
DATA WA LIKE SCARR.
DATA ITAB TYPE STANDARD TABLE OF SCARR.

START-OF-SELECTION.

 SELECT * FROM SCARR.
  MOVE SCARR TO WA.
  APPEND WA TO ITAB.
 ENDSELECT.

 LOOP AT ITAB INTO WA.
  WRITE: WA-CARRNAME, WA-URL(40).
  NEW-LINE.
 ENDLOOP.

END-OF-SELECTION.
```

```
REPORT  ZMX_07_03                    .

TABLES: SCARR.

DATA ITAB TYPE STANDARD TABLE OF SCARR
  WITH HEADER LINE.

START-OF-SELECTION.

  SELECT * FROM SCARR INTO TABLE ITAB.

  LOOP AT ITAB.
    WRITE: ITAB-CARRNAME, ITAB-URL(40).
    NEW-LINE.
  ENDLOOP.

END-OF-SELECTION.
```

Listato 3.

dall'esecuzione del codice.

Il trasferimento dei dati dal DB in una struttura interna per la successiva elaborazione, è talmente comune che ABAP dispone di scorciatoie per fare questa operazione massiva.

Esaminiamo il Listato 3, visibile in questa stessa pagina. La tabella ITAB viene dichiarata di tipo SCARR, pronta ad accettare i dati provenienti dalla SCARR, con la specifica "WITH HEADER LINE" dichiariamo che avremo bisogno di una area di lavoro dello stesso tipo, una sorta di "testata" della tabella o "record i lavoro".

Il punto di forza di questo esempio è la singola istruzione SELECT che carica direttamente l'intera tabella SCARR nella tabella interna ITAB. L'istruzione SELECT non è chiusa dalla ENDSELECT in quanto è una istruzione massiva unica che non rende disponibile un record set (parlando nel gergo comune ai linguaggi di programmazione che vanno per la maggiore oggigiorno).

Il successivo LOOP di utilizzo dei dati nella tabella interna ITAB va uso della "testata" per disporre di una area di lavoro nella quale si ritrovano i dati dei record letti all'interno dell'esecuzione del loop stesso.

Per proseguire sulla stessa lunghezza d'onda vediamo un esempio che carica in una tabella interna i dati provenienti da due tabelle del DB. In questo caso dobbiamo definire un record opportuno raggruppando dati dalla SCARR e dalla SPFLI, le due tabelle che abbiamo deciso di coinvolgere.

Il caricamento dlla tabella interna ITAB avviene in maniera "tradizionale", cioè leggendo i record dalle tabelle e preparando l'area WA che serve per caricare i record nella tabella interna.

Nella preparazione dell'area WA abbiamo usato l'istruzione "MO-

Figura 1.

Lufthansa	BERLIN	FRANKFURT
United Airlines	FRANKFURT	NEW YORK
United Airlines	FRANKFURT	SAN FRANCISCO
Qantas Airways	FRANKFURT	SINGAPORE
Lufthansa	FRANKFURT	BERLIN
Lufthansa	FRANKFURT	NEW YORK
Lufthansa	FRANKFURT	NEW YORK
Japan Airlines	FRANKFURT	TOKYO
United Airlines	NEW YORK	FRANKFURT
Lufthansa	NEW YORK	FRANKFURT
American Airlines	NEW YORK	SAN FRANCISCO
Delta Airlines	NEW YORK	SAN FRANCISCO
Delta Airlines	NEW YORK	FRANKFURT
Alitalia	ROME	FRANKFURT
Alitalia	ROME	TOKYO
Alitalia	ROME	OSAKA
Singapore Airlines	SAN FRANCISCO	SINGAPORE
Delta Airlines	SAN FRANCISCO	NEW YORK
United Airlines	SAN FRANCISCO	FRANKFURT
American Airlines	SAN FRANCISCO	NEW YORK
Qantas Airways	SINGAPORE	FRANKFURT
Singapore Airlines	SINGAPORE	SAN FRANCISCO
Singapore Airlines	SINGAPORE	JAKARTA
Singapore Airlines	SINGAPORE	TOKYO
Alitalia	TOKYO	ROME
Japan Airlines	TOKYO	FRANKFURT

VE-CORRESPONDING" che ci permette di muovere campi fra due strutture senza specificare il dettaglio di quali campi coinvolgere. L'istruzione si basa sul nome del campo che deve essere uguale nelle due strutture per essere coinvolto nella MOVE.

Le tabelle interne si possono ordinare con una singola istruzione:

SORT ITAB BY CITYFROM.

è ad esempio l'istruzione che ordina la tabella interna per città di partenza. La successiva elaborazione in LOOP stamperà i dati ordinati.

L'elaborazione risultante l'abbiamo mostrata nella figura 1 della pagina a fronte.

Prevedo anche da parte vostra la classica domanda che tutti gli studenti mi fanno arrivati più o meno a questo punto, e cioè: "si può fare una select con join invece che le select annidate?".

Si può ed ecco lo stesso codice del listato 4 nella parte di selezione fatta con la JOIN:

```
SELECT A~CARRNAME
       B~CITYFROM
       B~CITYTO
    FROM SCARR AS A
    INNER JOIN SPFLI AS B
      ON A~CARRID = B~CARRID
      INTO CORRESPONDING
           FIELDS OF WA.
    APPEND WA TO ITAB.

  ENDSELECT.
```

```
REPORT ZMX_07_04                    .

TABLES: SCARR, SPFLI.

TYPES: BEGIN OF RECORD,
        CARRNAME   LIKE SCARR-CARRNAME,
        CITYFROM   LIKE SPFLI-CITYFROM,
        CITYTO     LIKE SPFLI-CITYTO,
       END OF RECORD.

DATA WA TYPE RECORD.
DATA ITAB TYPE STANDARD TABLE OF RECORD.

START-OF-SELECTION.

 CLEAR: ITAB, WA.

 SELECT * FROM SCARR.
  MOVE-CORRESPONDING SCARR TO WA.
  SELECT * FROM SPFLI
   WHERE CARRID = SCARR-CARRID.
    MOVE-CORRESPONDING SPFLI TO WA.
    APPEND WA TO ITAB.
  ENDSELECT.
 ENDSELECT.

 LOOP AT ITAB INTO WA.
  WRITE: WA-CARRNAME,
         WA-CITYFROM,
         WA-CITYTO.
  NEW-LINE.
 ENDLOOP.

END-OF-SELECTION.
```

Listato 4.

Concludiamo qui questa settima puntata del nostro corso ABAP. Ci siamo addentrati in maniera definitiva nei meandri dei dati registrati sul DB e possiamo ora lavorare tranquillamente con le tabelle presenti.

Nella prossima puntata creeremo la nostra prima tabella e andremo a gestirne i dati attraverso i programmi che realizzeremo allo scopo.

[Mx]

Le prove di Jurassic News

Processor Technology SOL-20

Quando non si sapeva ancora come le schede a microprocessore potevano evolvere verso un completo sistema home, con terminale video e software di sviluppo, una piccola ditta, la Processor Technology, si trovò quasi per caso ad avere l'idea giusta.



Il SOL nella versione più diffusa, con cabinet in legno e metallo verniciato di blu.

Introduzione

SOL è un computer che appartiene alla prima generazione, quella che si colloca prima del 1980, diciamo prima dell'uscita dell'Apple II che secondo me è stato un vero spartiacque in questo senso. La sua costruzione si deve alle capacità tecniche di Lee Felsenstein e alla direzione di Bob Mash, che all'epoca era l'anima dell'Hombrew Computer Club. Dalla fucina di idee che venivano presentate dai soci del club, Bob sapeva trarre spunti e, cosa molto importante, trovare qualche soldo per realizzare i prototipi. Nella fattispecie il SOL viene attribuito alla Processor Technology, della quale Bob era fondatore. La Processor Technology era nata con l'intento di costruire e commercializzare schede di espansione per l'Altair 8800 ed aveva quindi il background giusto per affrontare la realizzazione di un calcolatore elettronico basato su una CPU e pochi altri componenti digitali. Il nome SOL deriva dal nome di Leslie Solomon, un articolista di Popular Electronics che lanciò la sfida per vedere se qualcuno era in grado di costruire un controller intelligente per il VDM-1, un videoterminale in grado di mostrare a video dei caratteri alfanumerici ma senza le funzioni di impaginazione, salvataggio, controllo, etc... che sarebbero state possibili con un computer dedicato.

Si stima ne siano stati venduti circa 10.000 esemplari, fra kit e as-

semblati, prima che la Processor technology fosse travolta dall'onda dello stesso successo del mercato che aveva contribuito a creare.

Primo approccio

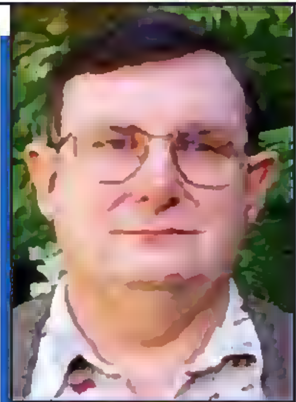
Il SOL si presenta con un cabinet singolo profilato per contenere la tastiera e un corpo centrale adatto a contenere le schede di espansione per il bus S-100, sul quale è basato. Per il SOL i progettisti hanno adottato un cabinet da amplificatore HI-FI (o meglio di un mixer) come peraltro da loro stessi dichiarato, basato su due fianchi in legno a sostegno di un guscio in lamiera formato da due pezzi: il sostegno alla tastiera, comprensiva di tastierino numerico (nella versione SOL-20), con una parte verticale sul fondo che alza il profilo della macchina di circa cinque centimetri e la chiusura del guscio sagomata a formare la base superiore e il retro dell'unità centrale. Il tutto ha l'aspetto di una "console" tipica delle sale di registrazione, anche se in versione molto ridotta.

La striscia verticale che stacca il piano della tastiera dal piano superiore è di un contrastante colore nero, mentre il resto della lamiera è verniciata in blu, sul quale viene ripetutamente ripetuta la frase "SOL Terminal Computer" sulla sinistra e un più discreto "Processor Technologic" sulla destra. L'uso alternato del colore arancio e bianco per le scritte, contrasta con lo sfondo nero e richiama subito all'occhio il logo del sistema. I fianchi in legno sono proprio "di legno", non come ci si aspetterebbe in multistrato o medio-density inpiallacciato: un segno del tempo che fu...

La tastiera è una tipica "teletype", disponibile anche con tastierino numerico, il feedback è meccanico e appare un po' "dura" per la nostra attuale sensibilità. Del resto questa senzazione è presente in tutti gli approccci alle macchine di calcolo fino almeno al 1985, se dotate di tastiera "vera".

L'alimentatore è interno e pertanto per rendere operativo il computer basta collegare il monitor e l'alimentazione. In relazione al tipo di sistema acquistato avviene il boot da ROM e ci si trova nel monitor del registratore a cassette (CUTS Computer Users Tape System) o nel sistema SOLOS se si dispone della costosa unità doppio floppy e software relativo.

La disponibilità di software per questa categoria di macchine era alquanto limitata. Già possedere un BASIC, si badi bene non grafico, era un lusso per gli appassionati, usi a combattere con l'assembler. Ora l'assembler va benissimo per scrivere brevi routine, anzi è proprio quello che ci vuole, ma quando



Questo signore è Lee Felsenstein, il "papà" del SOL Computer.



Il fascicolo del luglio 1976 di Popular Electronics, dove il progetto SOL ha fatto la sua apparizione.

viene voglia di sviluppare qualcosa di più complesso, anche una semplice rubrica indirizzi, il sorgente diventa prestissimo ingestibile.

Quindi i progettisti del SOL hanno puntato proprio sulla disponibiltà di un buon "monitor" di sistema, di un discreto BASIC e del lussuoso SOLOS per la gestione dei floppy. Con queste minime cose l'appassionato poteva dirsi felice, mentre riuscire a farci girare una versione di Chess da 1 Kb era una sfida da vincere piuttosto che l'obiettivo in grado di trasformare la macchina in un avversario per le partite a colpi di gambetto di donna.

Pet 2001 e SOL-20 a confronto in una foto presa ad una esposizione vintage di pochi anni orsono. Si apprezza una certa eleganza di proporzioni contro il massiccio cabinet della soluzione Commodore.



Specifiche tecniche

Il SOL computer è costruito attorno ad una CPU 8080A della Intel. Si tratta di un processore a 8 bit, diciamo di prima generazione, che necessita di un discreto set di chip di supporto e tre tensioni di alimentazione (+5, +12 e -12 Volts). La dotazione di RAM nei primi modelli è di soli 1 Kb, come la ROM contenente il monitor. Un'altro K di RAM è dedicato al video. Successivamente la RAM sarà espansa fino a supportare 48 Kb massimi e la ROM sarà integrata con moduli "applicativi", ospitati su una schedina intercambiabile.

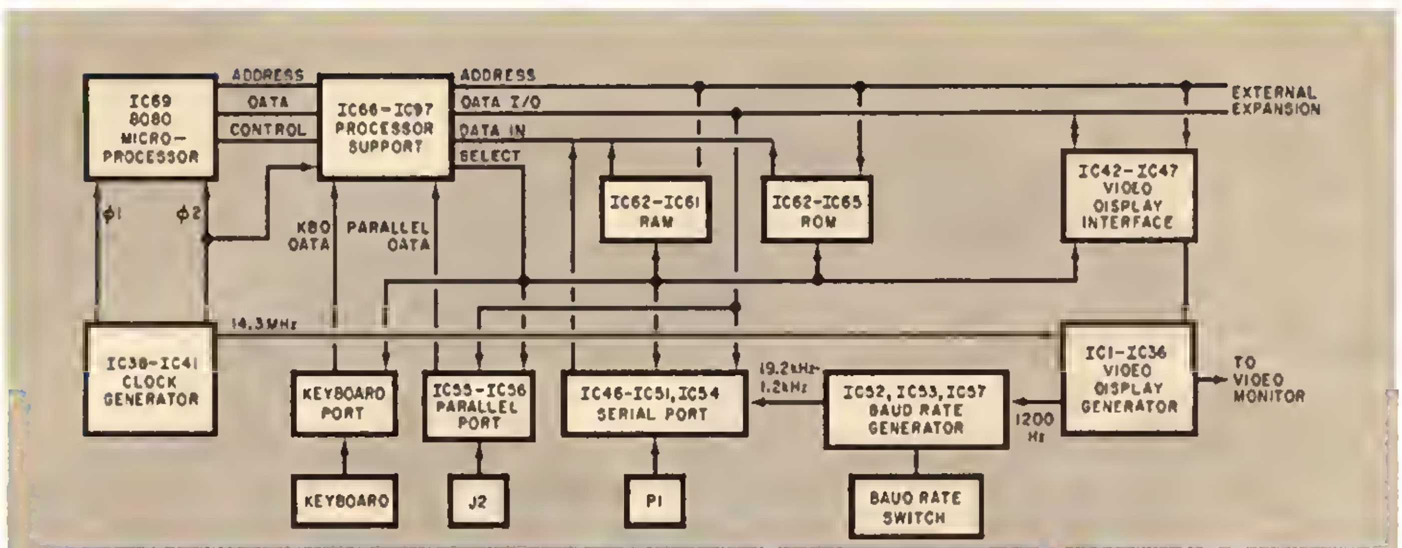
Il SOL nasce con una discreta dotazione di porte: una RS232 che è impostabile come current-loop, una parallela a 8 bit bidirezionale, interfaccia cassette, uscita monitor, slot per tastiera full-stroke e bus S-100 in grado di ospitare schede hardware basate su questo standard industriale.

L'output video non dispone di grafica e tanto meno di colore ma prevede un display inverso o lampeggiante e supporta un generatore di caratteri comprendente anche le minuscole. Sono possibili in output sedici righe da 64 caratteri ciascuna.

Hardware

Il SOL, come del resto le macchine dell'epoca, viene venduto con una ricchezza di schemi e manuali che farebbero impallidire oggi il rilascio di qualsiasi prodotto. Che si acquisti la versione in kit o quella assemblata, si hanno a disposizione gli schemi elettrici completi della macchina, le istruzioni per il montaggio meccanico ed elettronico e tutte le indicazioni utili a capire come funziona questo piccolo gioiello della tecnica.

Lo schema a blocchi è quello che diventerà classico: la CPU al centro con i chip di supporto (molti per l'In-

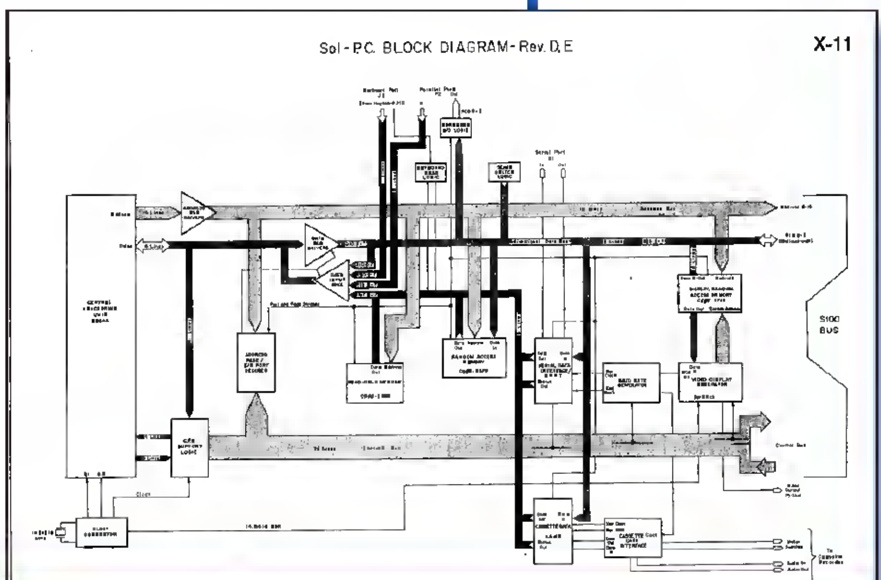


Schema logico (dalla rivista Popular Electronics).

tel 8080), i buffer di I/O, i bus dati e indirizzi e la memoria RAM/ROM. Colpisce la complessità del circuito che genera il clock, circuiteria che i processori delle generazioni successive ingloberanno nel loro die. Ovviamente molto complicato lo schema per la generazione del segnale video, anche qui non si disponeva di chip altamente specializzati e il segnale da mandare all'output veniva costruito pixel per pixel partendo dal contenuto della ram video scandita indirizzo per indirizzo dal generatore di caratteri.

Piccoli gioielli di tecnica anche gli schemi di controllo per la seriale e per il rimanente I/O, cassette deck compreso. Interessante anche esaminare la sorprendente complessità dello schema elettrico che soggiace alla codifica dei tasti della tastiera; ad onta della immancabile matrice decodificata che funge da ingresso, segue una complessa circuiteria in grado di restituire il codice del tasto digitato direttamente sul bus dati. Lo schema sembra dotato di bufferizzazione, almeno dell'ultimo carattere, ma i manuali non citano questo particolare e dovrei studiare meglio il circuito per esserne certo.

La doppia unità floppy chiamata HELIOS DISK II aggiunge sul bus S-100 il controller e la disponibilità di una promettente (e costosa) tecnologia per lo storage dei dati. Sui supporti da 8 pollici trovano spazio circa 380 Kb di dati, ma il controller, dato che può gestire quattro drive, porta la capacità in linea a circa 1,5 Mb che per l'epoca è una vera e propria manna. Ovviamente la capacità effettiva dipende poi dalla formattazione soft operata dal sistema operativo, che nel caso del PTDOS si attesta sui 270 Kb di disponibilità utente. L'unità a doppio



Lo schema funzionale tratto dal manuale in dotazione al computer.

floppy, assemblata e comprensiva di controller e cavetteria, costa nel 1977 la bellezza di 2700 dollari. Tradotto in lire con il cambio dell'epoca sarebbero poco meno di cinque milioni, più o meno un anno di stipendio di un impiegato medio.

La North Star, una ditta indipendente, offre una soluzione basata su floppy da 5,25": "Micro-Disk System" completa di controller e fino a 4 unità magnetiche, ciascuna formattata a 179 Kb.

La dotazione di periferiche ed espansioni si completa con una offerta di espansioni di RAM/ROM, interfacce di I/O sia standard che programmabili e periferiche principalmente legate alla locazione scientifica di laboratorio.

Software

Si è già detto della scarsa dotazione di programmi commerciali in grado di girare sul SOL. Bisogna quindi valutare la dotazione del software di base della macchina, vero valore aggiunto della soluzione Processor Technology rispetto ad analoghe iniziative hobbistiche in un settore, quello dei sistemi di calcolo a microprocessore, che cominciava alla metà degli anni '70 ad avere un discreto codazzo di appassionati.

Il sistema operativo, o più semplicemente il monitor di sistema, si chiama CONSOL OS (notare il gioco di parole con il nome della macchina).

L'intercambiabilità dei "personal module" permette tre scelte operative:

CONSOL: un monitor di sistema molto semplice che permtte il controllo delle operazioni principali come esaminare il contenuto della RAM, caricare la memoria video, etc...

SOLED: aggiunge le funzionalità di comunicazione a quelle base.

SOLOS: il modulo più avanzato, consente di usare il SOL come un vero e proprio computer personale.

Ci sembra opportuno parlare di quest'ultimo, il SOLOS e del sistema operativo per la gestione dei dischi, chiamato PTDOS (Processor technology Disk Operating System). Questi software di base rendono il sistema un vero calcolatore personale piuttosto che un semplice



Foto di gruppo. A sinistra del SOL un Oric-1, a destra un TSR-80 e un AIM-65.

terminale e quindi sono più vicini al nostro pensare e anche più interessanti. Inutile ripetere qui che all'epoca di uscita del SOL, ben pochi potevano vantare la conoscenza di un qualche sistema di controllo per macchine a microprocessore. Se si escludono gli specialisti elettronici, che disponevano di sistemi di sviluppo sofisticati e all'epoca dal costo proibitivo, i pochi appassionati che cominciavano ad interessarsi a questo strano marchingegno di macchina programmabile, dovevano imparare tutto ma proprio tutto.

Il SOLOS è disponibile tramite l'espansione "Personal Module" e, come dicevamo, dota la macchina di un set di comandi di programmazione e controllo, tutto in due Kbyte di memoria. Il set di comandi è a dir poco stringato in quanto si limita al controllo base della memoria, visualizzazone e modifica del contenuto, lancio del codice e controllo dell'I/O del registratore (eventualmente possono essere anche due collegati al sistema) per lo store/restore di un dump di memoria.

Il funzionamento della console SOLOS è il classico loop: prompt -> elaborazione -> prompt, con il prompt rappresentato dal carattere ">" e il Carraige/Line fead come tasto di fine input.

Uno dei tasti molto importante per il SOL è il Control-@, chiamato anche "Comand MODE" che in pratica è un break che interrompe qualsiasi attività in corso e restituisce il controllo al SOLOS, evidentemente generando un interrupt non mascherabile.

I comandi SOLOS si dividono in "famiglie" che ne orientano l'utilizzo. Troviamo i comandi "SET" che consentono di impostare i parametri base della macchina, i comandi "TAPE" che, come dice il nome, sono dedicati al registratore magnetico, e i comandi "CONSOLE" per l'interazione con la memoria e il processore. SOLOS ha una controparte batch chiamata CUTER, caricabile da cassetta, che trasforma il sistema in una console di controllo video a favore di eventuali apparecchiature esterne. Ad esempio se volessimo mostrare a video i dati di misura di una certa apparecchiatura, potremmo acquisire l'output della stessa, elaborarlo e renderlo a video impaginato, con scroll, lampeggiante, etc...

Al momento del reset il SOLOS è configurato per accettare comandi da tastiera e mostrare l'output a video. Tale configurazione di de-



In questa foto, a sistema aperto, si può apprezzarne l'ordinata disposizione delle piastre e l'ingegnerizzazione globale del sistema.

fault si modifica con una serie di comandi "SET". Ad esempio:

SET O=1 setta l'output verso l'interfaccia seriale

SET I=2 setta l'input alla porta parallela

0, 1 e 2 sono le tre periferiche "normali", cioé rispettivamente video/tastiera, seriale e parallela. Sono disponibili comunque tutte le porte custom nel range di indirizzamento del processore 8080, cioè fino a 2F, nel senso che l'utente potrebbe aver costruito una periferica di output interfacciandola ad esempio sulla porta 2E:

SET O=2E

sarebbe il comando per indirizzare l'output verso questa periferica.

Uno dei comandi SET permette di settare la velocità di comunicazione con i registratori:

SET TAPE n

(n=0 per 1200 boud, n=1 per 300 boud)

Il registratore audio non è una periferica molto veloce, come si capisce dal parametro di comunicazione. Del resto si tratta di fare il dump di pochi byte di memoria. A 300 boud si registra circa 1 Kb ogni trenta secondi.

Il controllo del registratore avviene tramite quattro semplici comandi:

GET (name(/unit) (addr)) Get a tape file into memory

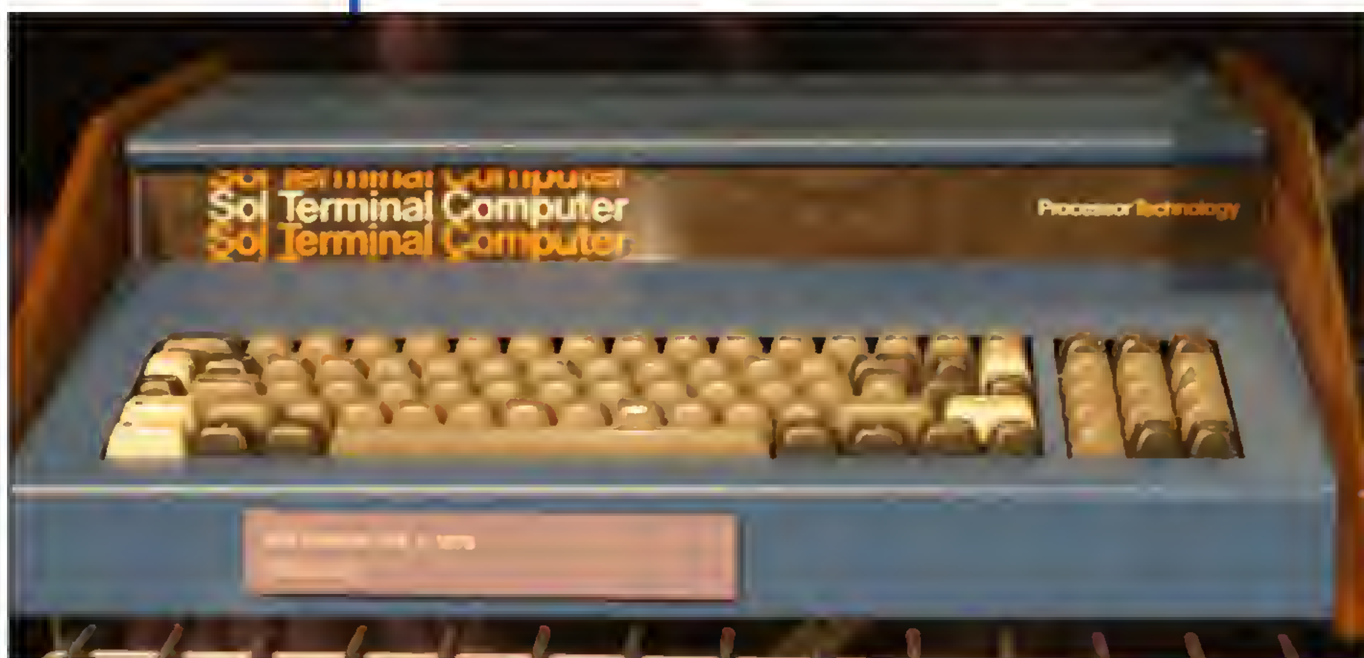
SAVE name (/unit) addrl addr2 (addr3) Save a file from memory to tape

XEQ (name(/unit) (addr)) Get then execute a tape file

CAT (/unit) Catalog tape files

il parametro /unit si riferisce al numero di unità coinvolta nel comando. L'unità 1 è quella di default, l'alternativa è l'unità individuata dal numero 2.

Il SOL esposto al Computer History Museum (California, USA). Un esemplare restaurato e curato dallo stesso progettista Lee Felsenstein che collabora anche ad altre iniziative di restauro finanziate dal museo stesso.





Ancora una immagina "da esposizione" del SOL accanto ad altre soluzioni più o meno risalenti allo stesso periodo.

E' sorprendente, visto con l'occhio di oggi, vedere la genialità con la quale i progettisti sono riusciti con soli quattro comandi a trasformare una periferica notoriamente poco flessibile, in una decente modalità di storing di dati e programmi. Con CAT si ottiene addirittura la directory (diremo oggi) della cassetta e ogni file può essere nominato con cinque caratteri e recuperato poi usandone il nome. Altra caratteristica è la possibilità di specificare l'indirizzo di caricamento del programma in memoria e la sua eventuale auto-esecuzione, anche da un indirizzo diverso (addr3).

Infine i comandi della console:

EXEC addr Begin program execution at 'addr'

ENTR addr Enter data into memory starting at 'addr'

DUMP addr1 (addr2) Dump memory data, 'addr1' to 'addr2'

TERM (portin (portout)) Enter Terminal Mode

CUST name (addr) Insert or remove a custom command

Interessante il comando CUST che permette una certa personalizzazione del sistema con la definizione di comandi custom:

CUST MYPRG 2F00

definisce un nuovo comando custom chiamato MYPRG che risiede in memoria e può essere mandato in esecuzione dall'indirizzo esadecimale 2F00.

Il funzionamento di questa feature è semplice: una tabella di comandi risiede in RAM, ogni comando CUST ne modifica il contenuto e la console lo esegue se non trova un corrispondente comando nel set standard.

EXEC, ENTR e DUMP sono i classici tre comandi che si trovano in tutti i "monitor" dei sistemi. Essi permettono rispettivamente di eseguire il codice da una certa locazione indicata, di modificare il contenuto della memoria a partire da un certo indirizzo e infine di visualizzarne il contenuto.

Il comando ENTR, seguito dall'indirizzo di partenza, rende diponibile un prompt di sistema (il carattere ":") con il quale si inseriscono in sequenza, separati da spazio, i byte che si intendono trasferire in RAM. Il carattere "/" termina la sequenza di input e ritorna al SOLOS.



Minimalista ma irreprensibile la copertina del catalogo della Processor Technology. Un libro dei sogni per gli appassionati del 1977!

Il comando TERM trasforma il SOL in un terminale che prende l'input da una certa periferica e lo manda in output. Sembra strano questo comando per un home computer ma non va dimenticato la particolare nascita di questo sistema, come controllore video, appunto.

La controparte batch del SOLOS è un programma chiamato CUTER che ne conserva i comandi ma in pratica salta la console interattiva. L'uso ci pare orientato agli specialisti, dato che la stessa sequenza di caricamento di CUTER da cassetta a memoria fa uso di uno speciale header di bootstrap. Non ne parleremo in questa sede in quanto ci sembra appesantirebbe troppo l'articolo senza svelare nulla di particolare sul sistema.

Come ogni "monitor" che si rispetti, anche SOLOS usa e mette a disposizione delle subroutine che possono essere usate in programmi utente. La lista di questi entry-point è ben documentata sullo user's manual. A parte le funzioni di base che ci si aspetta senz'altro di trovare in simili raccolte di codice, cioè input di un carattere da tastiera, output di un carattere a video etc..., si trovano anche gli entry point che controllano l'interazione con i registratori. Tramite esse è possibile, e il manuale ne riporta un esempio, programmare la registrazione di dati sulla cassetta, così come si userà fare con i file su disco.

A proposito dell'unità a disco, floppy in questo caso, il controller è programmabile con un ulteriore "salto di package", cioè utilizzando il sistema operativo PTDOS. Condizione indispensabile è ovviamente quella di aver equipaggiato il SOL con la doppia unità floppy chiamata Helios II Memory System e disporre di almeno 16 Kb di RAM.

Il sistema operativo PTDOS

PTDOS è un sistema operativo che conserva una sua vena "primitiva" nell'interazione con la macchina, nel senso che sono presenti alcuni comandi che pochi anni più tardi sarebbero stati giudicati "per veri specialisti", mentre molti comandi richiedono parametri che specificano indirizzi di memoria o ID di periferiche.

Per "boostrappare" la macchina con il PTDOS, che ovviamente risiede su un floppy, è necessario dotarsi del personal module chiamato BOOTLOAD, il quale aggiunge il comando BOOT al monitor di sistema SOLOS. Se ne può anche fare a meno, cioè disporre del solo SOLOS e usare il boot loader che

viene fornito in assembler in una appendice del manuale. Ovviamente sarà nostra cura salvarlo su cassetta una volta digitato, per non dover reinserire un'ottantina di byte ad ogni bootstrapp!

PTDOS è decisamente divertente ed è facile immaginare l'entusiasmo dei primi hobbisti che lo hanno avuto a disposizione. I comandi sono molti, ben documentati nelle oltre 160 pagine di manuale, e assieme al sistema operativo arrivano una serie di utility e di software aggiuntivi. Un esempio è l'Extended Basic, una versione potenziata dell'interprete, ma anche l'immancabile assembler e l'interprete FOCAL, di cui diremo più avanti.

PTDOS assomiglia abbastanza al CP/M, che sarà disponibile poco dopo, anzi una evoluzione del SOL-20 lo adotterà come sistema operativo opzionale, i file sul disco non hanno una struttura a directory ma sono "piatti", uno di seguito all'altro, come su un nastro magnetico. Del resto una scelta che non aggiunge una complicazione concettuale rispetto alla gestione del flusso dati su un nastro, organizzazione ormai digerita dai primi utilizzatori dei personal.

La flessibilità caratterizza il progetto di questo sistema operativo che, proprio per l'eterogenità delle periferiche disponibili, assomma caratteristiche molto interessanti. Ad esempio tutti i comandi sono reindirizzabili in input e in output, è possibile attivare un log di tutti i comandi ed eseguire dei macro oggetti che altro non sono che i ben noti file batch. Esiste il concetto di auto-esecuzione di un "programma" o macro al boot, la configurazione può essere modificata "al volo" con ben documentati comandi SET e si può impostare la protezione di scrittura per un range di memoria "bassa" definibile dall'utente.

La gestione dei file a disco è "dinamica", nel senso che i file, sequenziali o ad accesso random, si estendono alla bisogna e ogni device può essere pensato come un file nelle operazioni di I/O. Quest'ultimo è un concetto che sarà ottimamente sviluppato in Unix, ma ciò non deve sorprendere: le buone idee è sempre bene copiarle! Va detto che la realizzazione di device driver è anch'essa molto curata, segno evidente dell'attenzione che i progettisti mettevano nel realizzare un sistema che potesse estendersi con l'utilizzo delle periferiche più varie, a capriccio (e capacità tecnica) dell'utente. Per ultimo citiamo la disponibilità, anch'essa documentata, di far uso delle chiamate al sistema all'interno dei programmi utente.

La Processor Tecnology offre tre versioni del linguaggio BASIC, vero esperanto nell'epoca dei primi home. La ric-



La copertina del manuale utente dell'interprete FOCAL.



Un SOL-20 con doppia unità da 5,25 pollici.

chezza di istruzioni e la capacità di manipolare le varie struttura dati, matrici comprese, ne facevano derivare tabelle di confronto fra le varie piattaforme e non è escluso che si scegliesse il personal magari partendo dalla bontà della dotazione del BASIC disponibile.

La versione base, denominata Basic5, non manipola stringhe ed è adatta al primo approccio alla programmazione. Richiede peraltro la versione base della macchina. Il Basic8k è l'offerta di media complessità con la gestione delle stringhe, una dotazione di funzioni più ricca e la gestione delle periferiche, come il tape, da istruzioni nel linguaggio. Infine la versione "Extended" che offre un set di istruzioni completo e la capacità di lavorare con i file a disco, anche ad accesso random.

L'Extended Basic è una versione che potenzia l'interprete con istruzioni atte a considerare l'unità floppy come storage primario per l'immagazzinamento dei dati. In generale non è molto diverso dal classico interprete BASIC che conosciamo così bene.

Anche del Macro Assembler ci sembra inutile parlare, visto che offre, proprio per la natura del linguaggio, poche possibilità di personalizzazione. La dotazione di utility per l'assembler è completa: ci sono ben tre editor di testo, il più avanzato dei quali assomiglia molto ad un word processor, il compilatore (o meglio assemblatore), un comodo cross-reference e un discreto debugger simbolico, cioè che permette l'analisi del sorgente durante la sessione di debug.

Un passo in più per chi usa il linguaggio macchina come sorgente dei propri programmi, potrebbe derivare da un ambiente di utility che il SOL mette a disposizione sotto la sigla ASL8 (Assembler Source Language). Questi offre un fattivo supporto per chi deve sviluppare con l'assembler e gestirsi tutto l'I/O via interazione con le porte di ingresso/uscita assemblate nel sistema. Non ci pare il caso di procedere qui ad un approfondimento di questo environment, anche se una lettura del manuale ci ha lasciati sorprendentemente inclini a considerarlo molto utile per un programmatore "macchina".

FOCAL

E' del FOCAL (che starebbe per FOrmula CALculator) che ci sempra più interessante parlare, anche perché non si tratta di un linguag-

gio che ha poi avuto molta fortuna, anche se parecchie simil-realizzazioni.

Si tratta di un interprete orientato al calcolo numerico che deriva da una analoga realizzazione per i sistemi Digital. Si possono costruire programmi oppure comandare l'esecuzione automatica delle espressioni digitate; in entrambi i casi si possono utilizzare nomi simbolici per le variabili.

Un programma FOCAL è una sequenza di righe numerate come in BASIC ma con la particolarità di poter essere raggruppate tramite l'uso di un punto di separazione all'interno del numero di riga. Ad esempio:

```
: 2.0  SET A = 3.14 * 12.24
: 2.1  SET B = A - 2
: 3.0  SET RES = FSIN(B)
: 4.2  TYPE A, !, B, !, RE, #
: GO
```

Il codice definisce un programma di quattro statement che assegna dei valori calcolari alle variabili A, B e RES e ne stampa poi i rispettivi valori in output. I due punti all'inizio della riga rappresentano il prompt dell'interprete.

I valori numerici sono sempre floating point, ma sono possibili anche variabili alfanumeriche. Le funzioni, come FSIN nello statement 3.0, sono riconosciute dall'interprete proprio perché iniziano con la lettera F, proibita per le variabili. Nonostante FOCAL accetti nomi lunghi per le variabili, ne riconosce solo i primi due caratteri e addirittura i comandi per la loro sola iniziale. Cioè si sarebbe potuto scrivere la riga 2.0 così:

```
: 2.0 S A = 3.14 * 12.24
```

Per quanto appena affermato, la variabile RES si può nominare semplicemente "RE" nello statement di stampa. I caratteri speciali nello statement di stampa si riferiscono alla stampa di una riga vuota "!" e al semplice CR "#".

Il template delle istruzioni è fisso:

<numero di riga> <comando> <parametri>

dove <comando> è la parola chiave dell'istruzione, eventualmente ridotta alla sola iniziale.

I commenti usano il comando COMMENT o più semplicemente "C" all'inizio della riga.

Nelle istruzioni c'è il GOTO, il JUMP e l'IF, quest'ultimo ha come sintassi:

```
IF (expression) L1, L2, L3
```

che effettua il branch dell'esecuzione verso le righe L1, L2 e L3 rispettivamente se la valutazione dell'espressione fra parentesi è minore, uguale o maggiore di zero.

JUMP è più o meno come il GOSUB del BASIC ed effettua il banch fra più possibilità guidate dal valore dei parametri.

Il FOR, indispensabile istruzione per programmare loop, ha la se-

guente sintassi:

```
FOR I=A,B,C; <lista state-
ment>;
```

dove A è il valore iniziale della variabile di controllo (in questo caso I), B è il "passo", cioè l'incremento o decremento e C è il valore di chiusura. Seguono n statement validi del linguaggio con la sola limitazione che devono stare tutti, separati da virgole, su una stessa riga. Questa non è in realtà una grande limitazione perché con il DO si possono chiamare subroutine complesse quanto pare.

Esempio:

```
*20.2 S LX=40; S LY=70; S YN=0 S YX=100
*20.3 S XN=0; S XX=100
*20.5 S SX=(XX-XN)/LX ; S SY=(YX-YN)/LY
*20.6 A "DEFINE A,B,C FOR AX^2+BX+C" ,A,B,C
*20.7 T #,"GRAPH Y=AX^2+BX+C, X IS DOWN,"
*20.71 T "Y ACROSS"
*20.9 F X=XN,SX,XX; DO 60.0;
*20.95 Q
*60.1 S Y=X^2*A+(B*X)+C
*60.3 I (Y-YX) 60.5; S Y=YX; G 60.8
*60.5 I (YN-Y) 60.8; S Y=YN; G 60.8
*60.8 T "I"
*60.83 F J=YN,SY,Y; T "*";
*60.9 T # ;C RETURN
```

Questo esempio setta una serie di variabili (istruzioni 20.2 – 20.5), segue la richiesta di inputare il valore dei tre parametri A, B e C (istruzione ASK di riga 20.6) per il successivo FOR (riga 20.9). Viene richiamata la subroutine che inizia dalla riga 60.1 con l'istruzione DO, infine il QUIT (riga 20.95) termina l'esecuzione.

Per la gestione del sorgente troviamo solo tre statement: WRITE per avere a video (o su stampante, ricordate che il SOL è pienamente parametrizzabile in questo senso), ERASE per cancellare una riga o un range di righe e MODIFY per modificarla.

L'interazione con il sottostante PTDOS viene mediata dal comando LIB che permette di salvare e leggere i file su dischetto:

```
LIB SAVE myprog
LIB LOAD yourprog
```

Altro software

La presenza di un processore diffuso come l'8080, ha permesso al SOL-20 di acquisire in poco tempo una discreta dotazione di software, per lo più adattato da altre piattaforme. Le tipologie disponibili, oltre che coprire il comparto dello sviluppo, si concentrano verso il ludico (scacchi, star treck) e verso gli editor dei quali il SOL-20 può vantare un vero e proprio campionario. Si va dal semplice text editor fino al word processor chiamato Electronic Pencil (matita elettronica), un antenato del più famoso Word Star.

Terze parti hanno lavorato ad un offerta di hardware/software verticale per dotare il SOL-20 di interfacce, periferiche e ambiti applicativi (è stata sviluppata persino una scheda musicale con relativo software di gestione).

Rimane sempre la possibilità, peraltro molto incoraggiata nei primi

home, di costruirsi home-made il proprio software, magari con l'aiuto di riviste dedicate. Per il SOL ne registriamo tre, senza escludere che altre, soprattutto in forma di fanzine, possano essere esistite e circolate fra gli appassionati: si tratta di SOLUS, PROTEUS e PT ACCESS. Intendiamoci, niente copertina patinata, sono più fanzine che vere e proprie riviste, ma proprio perché costruite "sul campo" con le esperienze dirette degli utilizzatori, sono una utile e divertente miniera di informazioni e spunti operativi. Chi conosce le pubblicazioni del periodo ne riscontrerà una atmosfera di entusiasmo sorretta da continue scoperte e innovazioni, assieme a una costante ripetizione delle basi dell'informatica, bit per bit. L'informatica muoveva i suoi primi timidi passi fuori dall'ambito specialistico e cose che sembrano oggi banali: il bit, il byte,... erano anche difficili da assimilare.

Prezzi

Un listino prezzi della Processor Tecnology, datato 1977 e in nostro possesso, da una idea di quanto fosse costoso un sistema che poteva definirsi anche allora poco più che un oggetto di studio e hobby. Non è difficile immaginare che il costo al momento della produzione (1975) potesse essere addirittura superiore.

In generale si ha la prova di quella che possiamo definire "non popolarità" dell'oggetto. In fondo un personal computer interessava all'epoca pochi pionieri e qualche capitano di industria che cominciava a fiutarne le potenzialità in termini di marketing.

Il sogno di quelle poche ed intraprendenti persone che avevano trasformato il proprio hobby in una piccola azienda, era probabilmente quello di riuscire a vivere, possibilmente bene, del frutto del proprio ingegno.

I costi riflettono da una parte la bassa produzione dei chip e il costo del lavoro manuale necessario in tutte le fasi di lavorazione. Si tratta di una produzione di piccola serie, più artigianale che industriale.

La commercializzazione prevede tre configurazioni diverse per dotazione, denominate rispettivamente SOL System I, SOL System II e SOL System III. La versione base costa 1649 dollari in kit e 2129$ assemblato e comprende 8 Kb di RAM, il linguaggio BASIC e un monitor da 12". La versione intermedia porta la RAM a 16 Kb aumentando

MUSIC SYSTEM
USER'S MANUAL



Software sorprendente è stato progettato, in unione all'hardware, non appena si capì che il personal computer era veramente una "macchina universale".

il prezzo a 1883$ e 2283$, rispettivamente in kit e pre-assemblato. Infine il top, ma a che prezzo: 4750$ e 5450$, con 32 Kb di RAM, 2 drive floppy e BASIC avanzato per la gestione dei dischi.

A questa dotazione "base" è possibile aggiungere i famosi "Personal Module" con ROM e software vario che configura il sistema con diverse versioni di comandi fino a quello che assomiglia ad un vero sistema operativo, denominato PTDOS.

La Processor Technology commercializza anche la versione denominata SOL-10 o "Terminal Computer" che deriva dal primo prototipo ed è in pratica un single board computer con vocazione di video-terminale. Questo arriva a circa 1300$ assemblato, mentre se si vuole fare tutto da soli e comprare solo la mother board, si sborsano circa 600$.

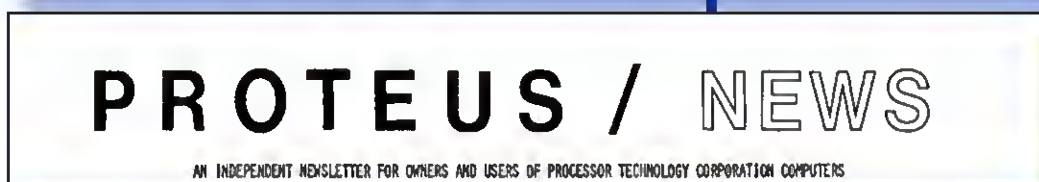
Un sistema "decente", cioè con buona dotazione di RAM e periferiche, l'indispensabile floppy e il software di base e di sviluppo, insomma il modello SOL-20/III, assomma un impegno monetario sui 5000 dollari (un dieci milioni di lire). Tradotti nella situazione economica italiana dell'80 si sta parlando di tre anni di stipendio di un operaio, ben oltre i tre milioni circa necessari a comprare una utilitaria.

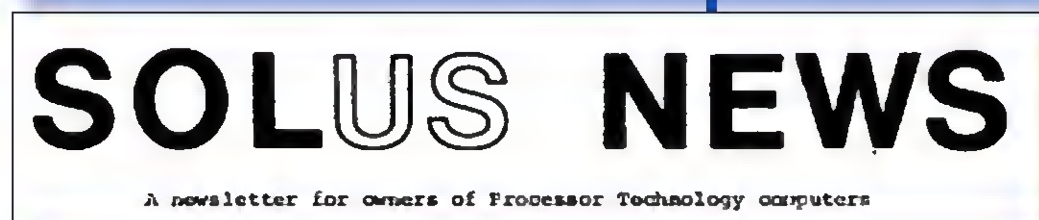
Il computer è decisamente un articolo di lusso!

Questo calcolo grezzo ci fa capire un'altra cosa per me particolarmente importante: come si poteva pensare che l'economia italiana potesse produrre innovazione e ricchezza, se per progredire nel caso specifico bisognava stare particolarmente bene di famiglia? E i primi passi in direzione di una innovazione tanto radicale non possono essere appannaggio dei centri di ricerca delle industrie: queste pensano a come incrementare il proprio business rendendo migliori i propri prodotti, ammazzare la concorrenza o progettare nuovi oggetti specifici del settore di mercato dove l'azienda opera.

Credo proprio che dobbiamo essere grati agli statunitensi se oggi possiamo goderne i frutti di una ricerca che non possiamo in alcun modo illuderci nel pensare di esserne stati protagonisti.







Le tre riviste/fanzine più note e forse piò diffuse, dedicate alla piattaforma di calcolo della processor Technology.

Conclusioni

Abbiamo esaminato un sistema che potremmo definire "spirito guida" nel senso di iniziativa che guardando avanti con rara dote visionaria rispetto a quello che poteva diventare il mercato mondiale dei sistemi di calcolo personali, ha raccolto le idee migliori che aleggiavano in quell'ambiente pionieristico che si identificava con il club Homebrew e ne ha saputo sintetizzare le idee in un oggetto, anzi "l'oggetto" per antonomasia, tanto atteso dagli appassionati.

Che si potesse fare una macchina in grado di offrire un output elaborato su un terminale video con l'elettronica a disposizione, può sembrare banale oggi, ma all'epoca non lo era affatto e questo è il grande valore di questo sistema.

Come hanno affermato ben più emeriti protagonisti, mancò solo una cosa a questo sistema per diventare veramente il riferimento vincente di un mercato che era solo agli albori ma che dopo pochi anni valeva già milioni di dollari. Questa cosa era semplicemente un cabinet decente, come quello che seppe sfornare la Apple per il suo][. L'idea che il sistema SOL-20 dovesse rimanere relegato al bancone di pochi appassionati, o tutt'al più ospite di qualche laboratorio universitario, e che quindi l'estetica non era cosa di cui preoccuparsi, è il limite non superato nelle idee dei progettisti.

[Tn]

TAMC

Teoria e Applicazioni delle Macchine Calcolatrici: la matematica e l'informatica, le formule e gli algoritmi, la completezza e la computabilità, le strutture dati e tutto quello che sta alla base dell'informatica.

La serie di Taylor

$$f(x) = \sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!}(x - x_0) + \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0)^{(n+1)}$$

Introduzione

Abbiamo visto nell'articolo sul calcolo della derivata prima di una funzione, che le capacità dei computer di rappresentare numeri piccolissimi (fino a 10^-308) è un'arma a doppio taglio. Infatti quando due numeri di grandezza troppo diversa sono coinvolti in operazioni matematiche anche elementari come la somma o la differenza, il risultato potrebbe essere inaffidabile.

Nella dissertazione presentata nell'articolo sul calcolo della derivata prima della funzione Seno(), abbiamo individuato in un'ordine di grandezza ^-16 quale limite superiore rispetto alla differenza fra due numeri per ottenere un risultato attenibile.

Il problema, lo ricordiamo è dovuto al fatto che ad un certo punto del calcolo è necessario aggiungere o sottrarre il valore dell'intervallo h alla posizione x0 dove si intende calcolare la derivata, secondo la formula che è la seguente:

$$f^{(1)}(x_0) = \lim_{h \to 0} \frac{f(x_0 + h) - f(x_0 - h)}{2h}$$

Quando h diventa troppo piccolo, cioè il suo ordine di grandezza si discosta più di 10^-16 da quello di x0 (che solitamente è unitario), il calcolo della somma o differenza fallisce per effetto della rappresentazione in virgola mobile presente nella macchina secondo lo standard IEEE-754.

La domanda spontanea è a questo punto: -"come fanno le funzioni di libreria a calcolare il valore delle formule con maggiore precisione?"

La risposta sta nell'approssimazione a polinomio della funzione da calcolare, cioè si cerca di scrivere un polinomio di un certo grado che approssimi la funzione con un errore controllabile. Il polinomio, essendo una funzione non trascendente, è sempre calcolabile.

Trattandosi di una approssimazione ci sarà sempre un errore, il segreto è rendere questa imprecisione piccola quanto possibile ma anche e soprattutto conoscendone l'entità.

Un metodo di polinomializzazione è conosciuto con il nome di Serie di Taylor, i cui elementi sono ricavati dalle derivate progressive della funzione, compresa la derivata "zero", cioè il valore della funzione calcolato nel punto X0. Cioè è come dire che in prima approssimazione la funzione si approssima al valore della funzione stessa in un "punto vicino", la seconda approssimazione fa uso del valore proporzionale alla derivata prima della funzione stessa, in terza approssimazione si aggiunge una correzione proporzionale alla derivata seconda e così via.

L'errore diventa sempre più piccolo mano a mano che alla serie si aggiungono termini. E' anche possibile che da un certo punto in poi i termini siano sempre a zero (succede approssimando un polinomio con la serie di Taylor, ammesso che questo abbia senso).

L'errore, che è quantificato dal secondo addendo della formula mostrata in apertura, è noto come resto di Peano e come si può facilmente notare dalla sua composizione è sicuramente più piccolo in termini assoluti dell'ultimo elemento calcolato nella serie. In questo resto compare un valore ξ *che è un punto scelto fra X e X0.*

Prendiamo in considerazione i primi due termini della serie (formula 1a). In questa e nelle formule che seguono abbiamo posto:

$$f(x) = \frac{f^{(0)}(x_0)(x-x_0)^0}{0!} + \frac{f^{(1)}(x_0)(x-x_0)^1}{1!} + O(h^2) \quad [1a]$$

$$f(x_0 - h) = f(x_0) + f'(x_0)h + O(h^2) \quad [1b]$$

$$f'(x_0) = \frac{f(x_0 - h) - f(x_0)}{h} + O(h) \quad [1c]$$

$$h = (x - x_0)$$

Si tratta di una approssimazione del punto di calcolo dell'errore che è peggiorativa, ma ci semplifica la vita permettendoci di affermare che l'errore è dell'ordine h^2.

dal momento che come è noto:

0! = 1! = 1 e che la potenza di zero di qualsiasi numero vale 1, possiamo scrivere con pochi semplici passaggi la formula [1b].

Noi vogliamo usare la formula di Taylor per calcolare la derivata prima e perciò abbiamo isolato la derivata prima ottenendo la formula [1c].

Quello che abbiamo ottenuto finora non è che ci aiuti moltissimo rispetto al problema dell'accuratezza del calcolo. Infatti rispetto alla formula di calcolo ricavata dalla definizione, abbiamo solo l'informazione sull'entità dell'errore, che

$$f(x+h) = f(x_0) + f'(x_0)h + \tfrac{1}{2}h^2 f^2(x_0) + O(h^3) \quad [2a]$$

$$-f(x-h) = -f(x_0) + f'(x_0)h - \tfrac{1}{2}h^2 f^2(x_0) + O(h^3) \quad [2b]$$

$$f(x+h) - f(x-h) = 2hf'(x_0) + O(h^3) \quad [2c]$$

$$\frac{f(x+h) - f(x-h)}{2h} = f'(x_0) + \frac{O(h^2)}{2} \quad [2d]$$

$$f'(x) = \lim_{h \to 0}(f'(x_0) + O(h^2)) \qquad [3a]$$

è proporzionale al valore h.

Ora coinvolgiamo il terzo elemento della serie, quello proporzionale alla derivata seconda. Possiamo ottenere due valori approssimati per le due quantità nelle formule [2a] e [2b].

La [2c] ci fornisce una quantità che ci serve per il calcolo della derivata prima secondo il metodo della differenza centrale. La novità sta nel fatto che questa quantità non contiene operazioni di somma o differenza da eseguire, possiamo evitare cioè il fastidioso problema della perdita di accuratezza dovuta alla limitazione della rappresentazione fp all'interno del PC. [3a]

Per passare dalla teoria alla pratica abbiamo preparato il programma in linguaggio C, visibile nella pagina a fronte.

Il risultato è la stampa della tabella mostrato sotto il codice dove si può apprezzare l'accuratezza del calcolo mano a mano che si aumentano le iterazioni, cioè aumentando l'ordine della derivata da utilizzare nella composizione della serie numerica.

Grazie al fatto che non ci sono più differenze fra grandezze piccole, si può spingere il segmento h fino a valori di 10^-56, un'enorme passo in avanti rispetto al valore di 10^-16 possibile applicando la formula diretta.

Conclusione.

Abbiamo fatto un piccolo excursus in quello che viene definito "calcolo numerico" che si occupa appunto di queste questioni di precisione, accuratezza e affidabilità dei calcoli. La lezione che ne ricaviamo è quella di non considerare il calcolatore un essere infallibile, ma tenere conto dei limiti intrinseci dovuti ai compromessi hardware-software cui i progettisti hanno dovuto adeguarsi. Questo quando si devono eseguire dei calcoli che non siano quelli banali della spesa di casa, ma che coinvolgono aspetti di progettazione magari legati alla stessa nostra sicurezza.

[Sm]

```
#include <stdio.h>
#include <math.h>

int main (int argc, const char * argv[]) {
    // insert code here...
      double h, x, x0;
      double cosx0;
      double derivata;
      int i;

      cosx0 = cos(0.5);


    printf("cos(0.5) = %.16f", cosx0, "\n\n");
      printf("\n");

      h = 0.5;
      printf("iter \tintervallo \t\t\tderivata \t\t\t\terrore\n");
      for(i = 1; i < 26; i++){
         derivata = (cos(0.5 + h) - sin(0.5)) / h;
         printf("%i \t %.16f \t %.16f \t %+.16f \n", i, h, derivata, derivata -
cosx0);
         h = h * 0.5 * 0.5;
         };
    return 0;
}
```

cos(0.5) = 0.8775825618903728

iter	intervallo	derivata	errore
1	0.5000000000000000	0.7240908924073870	-0.1534916694829858
2	0.1250000000000000	0.8453738746900736	-0.0322086872002991
3	0.0312500000000000	0.8699493187217406	-0.0076332431686321
4	0.0078125000000000	0.8757008882012940	-0.0018816736890788
5	0.0019531250000000	0.8771138150848117	-0.0004687468055611
6	0.0004882812500000	0.8774654797700805	-0.0001170821202923
7	0.0001220703125000	0.8775532978984302	-0.0000292639919426
8	0.0000305175781250	0.8775752463006938	-0.0000073155896789
9	0.0000076293945312	0.8775807330239331	-0.0000018288664396
10	0.0000019073486328	0.8775821046729106	-0.0000004572174621
11	0.0000004768371582	0.8775824476033449	-0.0000001142870278
12	0.0000001192092896	0.8775825332850218	-0.0000000286053510
13	0.0000000298023224	0.8775825556367636	-0.0000000062536092
14	0.0000000074505806	0.8775825649499893	+0.0000000030596166
15	0.0000000018626451	0.8775825798511505	+0.0000000179607778
16	0.0000000004656613	0.8775825500488281	-0.0000000118415446
17	0.0000000001164153	0.8775825500488281	-0.0000000118415446
18	0.0000000000291038	0.8775825500488281	-0.0000000118415446
19	0.0000000000072760	0.8775863647460938	+0.0000038028557210
20	0.0000000000018190	0.8775939941406250	+0.0000114322502522
21	0.0000000000004547	0.8775634765625000	-0.0000190853278728
22	0.0000000000001137	0.8774414062500000	-0.0001411556403728
23	0.0000000000000284	0.8789062500000000	+0.0013236881096272
24	0.0000000000000071	0.8828125000000000	+0.0052299381096272
25	0.0000000000000018	0.8750000000000000	-0.0025825618903728

L'intervista

Vari personaggi e amici incontrati qua e la per una chiacchierata sul mondo del retro computing.

Intervista a Marco (Voodoo)

Il retro computing si scopre a volte in posti impensati, come appunto racconto in questa storia di una scoperta e breve intervista, seguita ad un incontro fortuito avvenuto in una città che non è la mia e per puro caso.

Dunque ero per lavoro in questa città del nord Italia e mi dovevo fermare un paio di giorni. Passando in una via la mia attenzione fu catturata da una vetrina dove facevano bella mostra di sè un cabinet di pac-man, in funzione al 100%, quello originale della Namco, un flipper meccanico e una slot-machine, sempre meccanica. Questi oggetti non erano lì per destinazione, semplicemente costituivano l'arredo di una vetrina di una boutique, molto colorata e con un nome simpatico: Voodoo.

All'interno si intravvedevano ulteriori "marchingegni" d'epoca come un jukebox Wurlitzer 800 del 1940 (quello tutto colorato "a cupola") e un cabinet neintemeno del mitico gioco "Pole Position" della Atari (quello, se ve lo ricordate, dove si giocava seduti con tanto di volante e pedaliere).

Incuriosito più che mai e incoraggiato dalla presenza "rassicurante" di un uomo di mezza età all'interno al posto della classica commessa "bona, ma è tutto" come direbbe il mio amico Stefano, la quale è facile prevedere non mi avrebbe saputo dire nulla; entro e chiedo lumi, spiegando la mia curiosità e il mio interesse per questo particolare tipo di "merce".

Come era facile prevedere non si trattava di merce in vendita, ma dell'esposizione privata del proprietario che da un lato usava lo spazio di esposizione del negozio come magazzino e finestra pubblica a testimonianza di una tecnologia vintage e dall'altra ne ricavava una ambientazione insolita e divertente per la boutique, prevalentemente rivolta ad un pubblico giovanile (blue-jeans e giubbotti di pelle, insomma).

Marco, che chiameremo con lo pseudonimo di Voodoo, fra l'altro è quello che usa nel suo blog, mi ha concesso una interessante chiacchierata. Anche se l'incontro non ha potuto prolungarsi per i miei impegni, ne ho ricavato comunque una mezzoretta di piacevole conversazione con una persona intelligente e simpatica che ha risposto volentieri alle mie domande.

JN.

Come mai questa esposizione?

Voodoo.

Passione personale. Mi piace occuparmi di Vintage e in particolare di giochi che hanno qualcosa di meccanico o elettrico. Nel corso degli anni ne ho raccolti una buona quantità. Li restauro, vedo che funzionino e mi piace esporli qui e farci giocare i miei clienti.

JN.

Perché, come ho notato questi funzionano proprio. Ti confesso che sono entrato anche per vedere se era possibile farmi un giro di qualificazione in Pole Position :-)

Voodoo.

Certo che puoi! Li tengo apposta. Ho dei clienti che vengono qui regolarmente e mi propongono anche di mettere nuovi giochi. Poi naturalmente comprano qualcosa: faccio casa e chiesa, insomma!

JN.

Mi dici come hai cominciato, sì insomma, quale è stata la scintilla che ti ha fatto scattare questa passione?

Voodoo.

Sono sempre stato attratto dai giochi da Bar: i miei avevano un locale negli anni sessanta-settanta. Ho passato l'infanzia praticamente lì e quindi flipper e calcetto non avevano segreti per me.

Ma era proprio il flipper che mi attirava: l'idea di controllare la pallina e far scattare quelle carte per raccogliere i punti. La mia gioia vera era quando veniva la persona che faceva la manutenzione e ritirava l'incasso dei giochi. Sfilava il vetro e alzava il piano di gioco, una volta erano molto semplici e non avevano multi-piani. Sotto scoprivo un intreccio di fili, relè e bobine che mi lasciavano senza fiato dall'emozione!

Così anche per il jukebox, che all'epoca non mancava nei Bar. Il meccanismo di selezione dei dischi, con quella ruota che gira, il braccetto che li raccoglie e li posiziona sul piatto e poi la puntina che scende, tutto automatico, sincronizzato... Ne ero letteralmente affascinato!

Ad un certo punto il noleggiatore mise giù un jukebox nuovo che portava molti più dischi (96 per la precisione) ma si inceppava frequentemente. Il noleggiatore, stufo di venir chiamato un giorno sì e l'altro pure, mi diede le chiavi e mi spiegò (non ce n'era affatto bisogno), come sganciare la ruota e rimettere in funzione il tutto. Dopo un po' quando passava, non si occupava più di aggiornare i dischi nella macchina: semplicemente li lasciava lì, ritirava la percentuale e se ne andava. I dischi usati si usava lasciarli al proprietario del Bar. Ad un certo punto ho venduto tutta la collezione: erano più di 8000 dischi 45 giri!

JN.

Poi è stata la volta di Pong...

Voodoo.

Proprio così, ma intanto ero cresciuto e passavo i pomeriggi dopo scuola e tutta l'estate a lavorare per quello stesso noleggiatore. Nel locale dei miei genitori avevamo sempre l'ultima novità in fatto di giochi. Il flipper tirava ancora mentre il Jukebox lasciò il posto alla sequenza di cabinet che via via andavano per la maggiore.

JN.

E in questo "lavoretto" cosa facevi di preciso?

Voodoo.

All'inizio pulivo i giochi, facevo il giro dei posti con il titolare. Poi piano piano cominciai a intendermene ab-

bastanza e riuscire ad aggiustare i flipper in poco tempo. I jukebox, che si facevano sempre più rari a noleggio, erano tutti miei, nel senso che ormai me ne occupavo solo io, credo per il fatto che "costavo" pochissimo al titolare e rimettere in funzione una macchina del genere ci possono volere giorni.

JN.

E i giochi elettronici?

Voodoo.

Quelli mi attiravano un po' meno all'inizio: bisogna pensare che erano una tecnologia nuova e "chiusa" praticamente. Ti arrivava il cabinet già assemblato con una piastra enorme di circuiti che se si rompeva non c'era altro da fare che cambiarla in toto. Al limite arrivavamo a cambiare il joystick e i pulsanti: forse non hai idea di come si consumano i joystick quando la gente si affolla attorno al gioco per ore e ore! Ricordo che pulivamo la limatira di ferro sul fondo del cabine, limatira che era data dal consumo dell'asta del joystick.

JN.

Eravate in molti a fare quel lavoro?

Voodoo.

No, al contrario: pochissimi. Giusto il titolare e un ragazzo sulla trentina (io avevo appena sedici anni), che guidava il furgone e portava qua e la i giochi. Il titolare faceva la parte degli incassi, contratti e le riparazioni che richiedevano più esperienza.

Ricodo con tenerezza Damiano, così si chiamava il mio "collega". Era un ragazzo semplice ma furbo al punto giusto. A volte fingeva di non riuscire a riparare un certo gioco per poter tornare nello stesso locale in orario di servizio di una certa cameriera... Io non riuscivo a capire queste cose e ne ridevo: ero un ragazzo schivo e brufoloso, un classico adolescente scemo, insomma.

JN.

E il titolare?

Voodoo.

Quello era propio un furbastro: ho capito dopo che mi aveva sfruttato, nel senso che non ha mai regolarizzato la mia posizione lavorativa (erano altri tempi), però sapeva essere generoso e in un mese di lavoro estivo mi portavo a casa un milione tondo tondo.

Per l'epoca, attorno al 1974-75, erano una bella cifra ma il lavoro era duro: si lavorava ininterrottamente dalle nove di mattina alle nove di sera, sabato e domenica compresi. Sempre in giro fra un locale e l'altro passando dal lago alla montagna con alle calcagna le telefonate del titolare che lasciava messaggi in ogni posto dove dovevamo scendere. Sembra assurdo oggi che abbiamo i telefonini ma questa persona (preferisco non citare il nome) aveva inventato gli sms o i "pizzini" prima della Telecom o di Buscetta :-)

Veniva anche uno studente a darci una mano d'estate. E' stato lui, che frequentava la facoltà di ingegneria da qualche parte, a insegnarmi le prime cose sui microprocessori e a distinguere i chip che cominciavano a popolare anche i flipper.

JN.

E tempo per giocare ne avevi?

Voodoo.

Tantissimo: praticamente era il mio lavoro: giocare e giocare, per provare il gioco, naturalmente :-) C'erano anche giornate "buone" quando il titolare andava fuori città per procurare nuovi giochi o altri affari del genere. Se mi portava con sè era perchè aveva bisogno di compagnia e mi confidava cose anche personali delle quali ero im-

barazzatissimo, tipo i tradimenti che faceva alla moglie. Apparentemente, o almeno così pareva a me, aveva tresche dappertutto. Credo che molte cose fossero inventate: la moglie era un sergente e quando veniva in magazzino ci faceva mettere sull'attenti o poco meno! Ti spulciava tutte le note spese quando portavamo il conto del vitto a fine settimana. Dal momento che eravamo sempre fuori era inevitabile mangiare da qualche parte. Lui era molto generoso e ci diceva sempre: -"Ragazzi, fermatevi a prendere una birra ogni tanto!". Damiano non se lo faceva dire due volte e credo che se ci fosse stato l'alcool test sarebbe finito più di una volta in trattamento coatto obbligatorio per disintossicarsi. Io ero quello della coca-cola; mi piaceva proprio e poi era di moda, ti faceva sentire "ganzo" quando giravi con la bottiglietta di vetro in mano tracannando quell'intruglio dolciastro. Ora non riesco a berne nemmeno un sorso: credo sia troppo zuccherata per i miei gusti attuali.

JN.
E i computer?

Voodoo.
Non ne sentivo il bisogno: i giochi da Bar erano innumerevolmente più avanzati rispetto a quanto vedevo sugli schermi degli amici che avevano il Commodore 64 o qualcosa di simile. Naturalmente poi ho comprato il PC e cominciato a giocare qualche titolo, ma ormai il "momento magico" era passato e poi non riuscivo ad appassionarmi ai nuovi standard. Per me lo stile dig-dag era l'ideale: prendi il joystick e vai, non sono mai riuscito a farmi piacere i giochi di strategia.

JN.
E perché hai cambiato completamente settore di attività? Non hai pensato di metterti a fare il noleggiatore per conto tuo?

Voodoo.
Ecco, magari sarebbe successo, ma i miei genitori non erano convinti fosse una attività "onesta" al 100%. Dicevano che eravamo degli "zingari", senza offesa per quest'ultimi, semplicemente sempre in giro con questo furgone, a guisa di persone "senza fissa dimora". Un po' avevano ragione, poi c'è stato il patacrack della figlia del titolare (non cito il nome: il paese è piccolo).

JN.
Cioè?

Voodoo.
Bhe, eravamo giovani entrambi: io sui diciotto e lei appena quattordici o quindici. Fu lei che cominciò a venire in magazzino "per fare una partita" diceva, ma poi da cosa è nata cosa e un giorno sua madre entrò come una furia e letteralmente mi sbattè fuori. Ti giuro che non eravamo andati aldilà della pomiciatina, e credo non sapessi nemmeno cosa veniva dopo... ma la madre mi fece sentire peggio di un ladro!
Lui, il padre intendo, non si fece vivo per niente, venne una sera nel locale dei miei, si prese i giochi, mi lasciò quello che secondo lui era "quello che mi aspettava" e non lo rividi più se non casualmente dopo dieci anni. Stavamo al pronto soccorso, lui con non so che male e io con una caviglia fuori uso. Ci scambiammo dei saluti, più dei convenevoli che altro, e questo è tutto.

JN.
E la ragazza?

Voodoo.
Boh, francamente non so che fine abbia fatto. Penso che se la incontrassi non la riconoscerei di certo. Io ero un ragazzo e lei poco più di una bambina. Però ebbi una

lezione di vita da questa storia.

JN.
E cioè?

Voodoo.
Vedi, negli ultimi tempi passati dal noleggiatore mi ero convinto di essere indispensabile alla ditta. Questo mi ringalluzziva, mi sentivo un Dio, se puoi capirmi. Veniva gente dalle città vicine per farsi modificare dei giochi vecchi e, copiando le Eprom, riciclare l'hardware. Quello che mi sorprese è che il tizio, il mio ex datore di lavoro insomma, potesse permettersi di mandarmi via dall'oggi al domani. Non può andare avanti, pensavo, dovrà chiudere baracca. Invece non successe proprio nulla, la ditta continuò la sua attività e il baldo giovane smanettone fu solo una meteora, una luce intensa ma futile che brucia tutta la materia e arriva a terra al massimo un sassolino insignificante, quando magari è partita da tonellate e tonellate di roccia!

JN.
E Damiano?

Voodoo.
L'ho rivisto qualche volta: era sempre lo stesso, faceva la medesima vita, sempre lo stesso orario giorno dopo giorno, sette giorni su sette d'estate con una settimana o poco più di ferie all'anno. Credo che avesse mesi di ferie arretrate, ammesso che il titolare l'avesse assicurato al 100%, cosa di cui dubito. Poi le nostre strade si divisero del tutto, io cominciai un'altro lavoro presso dei parenti che avevano un magazzino all'ingrosso di vestiario, ecco perché ora mi trovo qui.

JN.
La fine di una brillante carriera di tecnico elettronico.

Voodoo.
Proprio. Credo che mi sarei buttato nel ramo: il mercato tirava (attorno agli anni '80), i giochi da Bar facevano incassi inimmaginabili: se indovinavi il posto e il gioco, cose non facile, in un mese si ripagava tre volte e ti rimanevano ancora quasi tre anni di vita da sfruttare. I giochi nuovi si mettevano dove c'era folla di giocatori: nei locali sul lago d'estate ad esempio, poi si spostavano pian piano "in periferia". I locali più sfigati si beccavano sempre le terze scelte.

Nell'ultima stagione il titolare mi fece una proposta: avrei dovuto pagarmi con l'incasso di due locali: quello dei miei e un'altro che non era distante.

Nonostante la fiacca estiva riuscì a mettere assieme una cifra tale da comperarmi la mia prima auto. Era una Fiat 127 rossa che mi costò quattromilioni e trecentomila lire, tutto il guadagno della stagione, ma niente male, ti pare? L'anno successivo dovetti arrangiarmi con un altro lavoretto estivo con il quale ricavavo poco più di mezzo milione al mese.

JN.
Ma comunque a giudicare da quel che vedo non sei mai uscito del tutto dal settore.

Voodoo.
Quando i miei vendettero la licenza del Bar mi comperarono questa boutique: il lavoro è migliore e mi lascia molto più tempo per me. Casualmente conobbi una persona che cercava qualcuno che gli restaurasse un Rock-Ola, una marca di jukebox molto in voga negli anni '60. Mi offri di farlo e sorprendentemente la cosa mi piacque molto. Così rientrai nel giro degli appassionati frequentando fiere e mercatini e recuperando materiale dai depositi di questi noleggiatori che ormai facevano solo videogiochi. La voce girava e cominciai a ricevere tele-

fonate da tutta Italia di gente che mi proponeva scambi, recuperi o cercava un restauratore.

Oggi lavoro alla boutique nel pomeriggio, mentre la mattina la dedico tutta al mio hobby (non tutti i giorni, ma quasi).

JN.

Mi puoi fare una tua personale classifica dei giochi? Quali preferisci e quali sono stati i migliori di tutti i tempi?

Voodoo.

Credi a me, l'apice del coinvolgimento ludico è stato raggiunto con il bigliardino elettromeccanico, il flipper, come lo chiamiamo in Italia (e in nessun'altra parte del mondo, pare). Il contatto fisico della macchina era palese: la tenevi in pugno, la accerezzavi e poi quel movimento così "fallico" della spinta che i più scafati sanno fare con vera maestria...

Ricordi quella serie TV con Fonzie?

JN.

Happy Days...

Voodoo.

Proprio. Arthur Fonzarelli ha incarnato la vera anima del giocatore di flipper: spaccone ma solo con i pulsanti, duro ma solo con i respingenti, con movenze feline sapeva cogliere quella che io chiamo "la vera anima del flipper"!.

E' inutile che ti dica i nomi di questi gioielli della Bally, non sarebbero significativi dal momento che non sono conosciuti come i videogame.

Per quanto riguarda i giochi da Bar direi che ho delle preferenze per qualche titolo, ma sono parzialità passeggere. Magari se ne trovo uno nuovo e mi metto a restaurarlo, poi mi piace giocarci per mesi magari, poi passo ad altro.

JN.

Però vedo che hai messo Pac Man in vetrina.

Voodoo.

Questo perché lo conoscono tutti e quindi fa più "vintage". Il mese scorso il suo posto era occupato da Space Invaders della Namco, sicuramente lo conoscerai: quei marziani che scendono in file compatte come da una scalinata. Sono molto ingenui questi titoli, ma si era solo all'inizio...

JN.

E perché non hai chiamato il tuo negozio "JukeBox" o qualche cosa del genere?

Voodoo.

Ci avevo pensato e ho scartato decine di nomi, compreso "PacMan" perché credevo non si potesse usare, visto che è un marchio registrato. Poi in un cassetto della scrivania ho trovato una vecchia scheda video per PC: Voodoo.

JN.

Sì ricordo, era un accelleratore grafico usato attorno al 1995, mi sembra di ricordare.

Voodoo.

Infatti, così non ho cercato oltre. Il nome è abbastanza esoterico e generico al punto di non essere oggetto di recriminazioni. Poi questa cosa della magia, i misteri, i fantasmi, etc... sono altre cose delle quali mi sono interessato, anche se superficialmente.

Sei stato molto gentile e ti ringrazio tantissimo della chiacchierata.

Voodoo.

Non c'è di che, torna quando vuoi. Anzi se mi avvisi qualche giono prima possiamo organizzare e ti porto a vedere il mio magazzino dove tengo le macchine, è un po' fuori città, ti interessa?

JN.

Retro Code

Il vecchio codice rivisto dall'alto di una evoluzione durata trent'anni.

BASIC Conversions (1)

La conversione dei programmi BASIC fra Apple II e TRS-80.

Premessa.

Affronteremo in questa serie le tecniche di conversione dei programmi BASIC fra le macchine di maggiore diffusione apparse agli inizi degli anni '80.

Il contesto è ben noto: i programmi si creavano o si copiavano direttamente da tastiera o si scambiavano fra utenti di una stessa piattaforma, con i supporti disponibili (prima le famigerate cassette audio, poi i floppy). La diffusione del calcolatore non era però così vasta, anzi ancora nel 1985 ricordo che la percentuale di studenti universitari che possedevano un proprio calcolatore si aggirava attorno al 10% e io frequentavo una facoltà scientifica; non azzardo neppure una stima in altre facoltà... Oltre a questo si doveva mettere in conto l'incompatibilità delle piattaforme e la stessa scarsità di documentazione tecnica.

Ecco che allora la possibilità di convertire un sorgente BASIC da una piattaforma all'altra era quantomeno interessante e perseguita qualora ci si imbattesse in un sorgente particolarmente utile.

Perché convertire?

Come si è appena detto, quando ci si imbatteva in un sorgente scritto per una certa piattaforma che appariva interessante per il nostro lavoro e/o studio (ma anche per il semplice divertimento), spesso la strada di rifare tutto da capo copiando l'idea o quella di tentare una conversione del sorgente, erano le uniche strade disponibili. Se poi si mette in conto che il BASIC non è proprio il massimo per leggibilità, la scarsezza dei commenti nel codice (rubano spazio) e il comportamento a volte "fantasioso" di certi statement BASIC, senza contare i PEEK&POKE, si capisce come la strada della conversione fosse irta di difficoltà.

Rivedere oggi questi concetti, del tutto obsoleti siamo d'accordo, ci serve per fissare un "sapore culturale" che fa parte anch'esso, al pari di altri nobili argomenti, della storia dell'informatica personale.

Le prime due piattaforme che trattiamo sono da una parte l'Apple II, macchina diffusissima ancorchè costosa e il Tandy Radio Shack serie 80, diffusa in pari misura oltre oceano (molto meno in Italia per mancanza di un importatore ufficiale). Già il fatto che si tratta di due piattaforme che usano un diverso mi-

croprocessore (il 6502 per Apple e lo Z80 per il TRS-80) ci dice che la conversione potrebbe non essere tanto semplice.

Per quanto riguarda l'Apple II, si parla dell'interprete Applesoft, mentre per il TRS-80 si dovrebbe tenere conto che ne esistono due modelli caratterizzati da differenze significative di architettura (Model I e Model III).

I due BASIC non sono diversissimi per fortuna. Nella tabella 1 ne diamo un resoconto per quanto riguarda le istruzioni da modificare con un breve commento.

E' evidente che ci stiamo concentrando sulle funzioni di calcolo e di esecuzione "pura" del sorgente. Se affrontassimo la conversione di suoni e grafica le cose non sarebbero tanto semplici. Per quanto riguarda la conversione delle istruzioni grafiche in particolare è quasi impossibile ottenere una conversione uno-a-uno fra i due sorgenti e conviene invece riscrivere le parti grafiche dopo averne compreso in pieno il funzionamento.

La lista delle istruzioni che si possono trasferire pari-pari è invece quella che è scritta nella tabella 2.

Tabella 1 (continua nella prossima pagina). Lista dei comandi (istruzioni che hanno una sintassi differente.

APPLE II	TRS-80	Commenti
CHR$(7)	na	Emissione di un Beep dallo speacker di sistema. In una conversione si può rinunciare all'istruzione
CLR	CLEAR	Istruzione non usata nell'Applesoft, mentre si trova nell'Integer BASIC (la versione senza virgola mobile)
CON	CONT	Continuazione dell'esecuzione dopo uno STOP
DEL x DEL x,y	DELETE x DELETE x,y	Cancellazione della riga x del sorgente o del range di righe da x a y
GET x$	INKEY$	Accetta un tasto dalla tastiera. mentre l'istruzione GET di Apple si ferma ed assegna alla variabile x$ il tasto premuto, nel TSR-80 l'istruzione non si ferma ed inoltre deve essere gestita l'assegnazione. Lo statemente seguente realizza sul TRS-80 le stesse funzionalità dell'Apple: 10 IF INKEY$ = "" THEN GOTO 10
HOME	CLS	Cancella il video e posiziona il cursore in alto a sinistra
HTAB(x)	PRINT @ o TAB(x)	Posizionamento del cursore sulla riga

APPLE II	TRS-80	Commenti
LIST, x	LIST=x	Lista il sorgente dalla riga x fino alla fine del programma; si tratta dell'unica variante dell'istruzione LIST diversa fra le due macchine, tutte le altre hanno la stessa sintassi
LOAD	CLOAD	Caricamento da cassetta
NOTRACE	TROFF	spegnomento della funzione di TRACE durante l'esecuzione del programma
ON ERR GOTO	ON ERROR GOTO	Stessa istruzione, provoca il trasferimento dell'esecuzione al verificarsi di una condizione di errore
PLOT x,y	SET(x,y)	Accende un pixel nella posizione individuata dalle coordinate x (orizzontale) e y (verticale)
RECALL x	INPUT #-1	Rilettura di un dato da cassetta. Il comportamento non è esattamente lo stesso fra le due piattaforme
RND(-x)	RANDOM	Genera un numero casuale che è sempre lo stesso ad ogni esecuzione
SAVE	CSAVE	Salvataggio del sorgente su cassetta
SCRN(x,y)	Point(x,y)	Restituisce il colore del punto di coordinate x e y
SPC(x)	STRING$(x, 32)	Inserisce x spazi dopo l'ultima istruzione PRINT
STORE x	PRINT #-1	Scrive una variabile o un array di variabili fp su cassetta
TRACE	TRON	Accende la funzionalità di tracing del sorgente durante l'esecuzione
VTAB x	PRINT@ x	Posizionamento verticale del cursore di stampa

Oltre alle istruzioni che non richiedono modifica (Tabella 2) e quelle che necessitano di accomodamenti (Tabella 1), ci sono una serie di istruzioni disponibili sulla piattaforma Apple che non trovano possibilità di conversione sul TRS-80. Queste sono tutte le istruzioni grafiche (o quasi tutte, eccetto SET e POINT) ed alcune altre legate all'hardware. Queste le abbiamo elencate per completezza nella tabella 3.

Che fare se ci si trova difronte a istruzioni specifiche dell'Apple II

STATEMENT COMUNI	STATEMENT COMUNI	STATEMENT COMUNI
ABS(x)	INPUT A$	RETURN
AND	INT(X)	RESUME
ASC(X)	LEFT$(X$, X)	RIGHT$(X$, Y)
ATN(X)	LEN(X$)	RND(X)
AUTO XXX	LET	RND(0)
AUTO XXX,YYY	LIST	RUN
CHR$(X)	LOG(X)	RUN X
CLEAR	MID$(X$, X, Y)	SGN(X)
CONT	NEW	SIN(X)
COS(X)	NEXT	SQR(X)
DATA X,Y,Z	NOT	STEP X
DIM X$(Y)	ON X GOSUB xx,yy,zz	STEP -X
END	ON X GOTO xx,yy,zz	STOP
EXP(X)	OR	STR$(X)
FOR X=Y TO Z: NEXT	PEEK(X)	TAB(X)
FRE(X)	POKE X, Y	TAN(X)
GOSUB X	POS(0)	USR(X)
GOTO X	PRINT	VAL(X$)
IF ... THEN	READ X	
INPUT X	REM	
INPUT X,Y,Z	RESTORE	

Tabella 2. Lista statement comuni alle due piattaforme. Non occorre nessuna conversione.

che dobbiamo trasferire sul nostro TRS-80?

Lasciando perdere quello che riguarda la grafica, che implicherebbe una riscrittura totale, ammesso che si possa fare, viste le grandi differenze in questo senso sulle due piattaforme, ci troviamo difronte in pratica a tre possibilità:

1. ignorare del tutto l'istruzione, quando questo è possibile;

2. modificare l'istruzione con una equivalente che faccia la stessa cosa o una simile, ma accettabile;

3. sostituire allo statement Apple una funzione o una subroutine creata apposta per risolvere il problema.

Prendiamo il caso numero 1. E' evidente che una attività di conversione, soprattutto di un programma applicativo (non di un gioco grafico ad esempio), può fare a meno di tante istruzioni "di contorno" che hanno, in questa tipologia di codice, più che altro funzione di abbellimento.

Ci riferiamo alle istruzioni che tracciano il testo a colori o in modalità flash o inversa. Nel caso dell'Apple II inoltre esistono una serie di "pseudo-istruzioni", diremo così, realizzate con POKE che vanno ad incidere sulla visualizzazione. La modalità TEXT dell'Apple II è infatti molto flessibile e non è raro trovare programmi che facciano uso di un POKE 22, n per settare la larghezza delle righe di testo.

NON DISPONIBILI SU TRS-80	
CHR$(7)	beep del terminale
COLOR=x	setta il colore per la prossima istruzione grafica
DEF FN x	definizione di funzione utente
DRAW x	tracciamento linea grafica
FLASH	attributo flash dell'intero screen
HCOLOR=x	setta il colore in alta risoluzione
HLIN x, y	tracciamento di una riga grafica
HPLOT x,y	tracciamento linea in alta risoluzione
IN# x	input dalla periferica nello slot x
INVERSE	tracciamento testo in modalità inversa
LOMEM	definizione area riservata per grafica
NORMAL	stabilisce la modalità normale del testo a video
PDL(x)	istruzione riservata all'interazione con le "paddle", joystick analogico
PR#x	setta l'uscita di sistema sulla periferica nello slot x
ROT	istruzione grafica
SCALE=x	scalatura di una SHAPE
SHLOAD	carica una bitmap da cassetta
SPEED=x	modifica la velocità di display dei caratteri a video
TEXT	setta la modalità testo uscendo da quella grafica
VLIN x, y	tracciamento linea verticale
WAIT x,y,z	stop del programma in esecuzione in attesa del verificarsi di una certa condizione

Tabella 3. Lista delle istruzioni dell'Apple che non trovano corrispondenza nella piattaforma TRS-80.

La completa trattazione di questo argomento esula dallo scopo dell'articolo per cui rimandiamo ad un apposito pezzo, magari nella rubrica Apple Club, la desamina delle locazioni di memoria della macchina di Apple Computer.

Le situazioni che rientrano nel caso numero 2, cioè la sostituzione con equivalenti "similari" di certe istruzioni riguarda pochi casi nei quali esiste una istruzione o un comportamento del TRS-80 simulabile sull'Apple con chiamate allo routine del monitor.

Dal momento che mancano istruzioni specifiche, i listati BASIC Applesoft riportano spesso istruzioni del tipo:

CALL -936

Questa in partilare equivale all'istruzione CLS del TRS-80. Una lista delle più comuni l'abbiamo riportata nella tabella 4.

APPLE II	Equivalente TRS-80
CALL -936	CLS Cancella i caratteri della riga e riporta il cursore all'inizio della riga stessa
CALL -958	PRINT CHR$(31) cancella i caratteri dalla posizione di stampa fino alla fine del video
CALL -868	PRINT CHR$(30) cancella i caratteri a destra del cursore
X = PEEK(-16384)	A$=INKEY$:X=ASC(A$) + 128 lettura di un carattere dal buffer di tastiera
POKE -16368, 0	pulizia del buffer della tastiera. Non esiste un equivalente TRS-80 ma si può simulare con un loop di INKEY$
X=PEEK(36)	X=POS(0) ritorna la posizione orrizzontale del cursore

Tabella 4. Funzionalità comuni con l'equivalente istruzione TRS-80.

Il caso 3 riguarda quelle istruzioni Apple che non sono facilmente traducibili, in particolar modo la grafica, anche a bassa risoluzione. L'Applesoft da questo punto di vista è molto flessibile comprendendo tre modi grafici: il classico TEXT, la bassa risoluzione, chiamata GR, la HGR alta risoluzione e un modo misto che gestisce una pagina grafica e una finestra di testo nella parte bassa del video.

Se ciò non bastasse ci si mettono di mezzo anche le duplicazioni delle pagine grafiche e conseguente scambio per la visualizzazione.

La traduzione in questi casi è improba e non facilmente standardizzabile.

Concludiamo qui questa prima parte che ci ha portati a scoprire da vicino quali differenze esistevano dei due interpreti BASIC di due delle macchine maggiormente diffuse nei primi anni 80: l'Apple II e il Tandy Radio Shack-80. Chissà se qualcuno ci ha provato a scrivere qualcosa di automatico ma ne dubitiamo molto, vista la mole di codice necessaria al riconoscimento delle varie situazioni.

[Tn]

Laboratorio

Interventi hardware e software per il ripristino e l'evoluzione dei sistemi di calcolo personale.

La conservazione della carta

Introduzione

Molto del materiale relativo al retro computing è di carta. Sembra assurdo, eppure è proprio così: manuali, libri e riviste fanno parte stabilmente delle nostre conoscenze e diventano essi stessi delle collezioni da portare avanti, completare e soprattutto conservare.

Sull'opportunità di procedere ad una generale scannerizzazione del materiale e quindi affidare la conservazione ai media di storage, sappiamo che non sempre è una buona idea. C'è poi il problema della mole di questi dati e di conseguenza dello spazio che andrebbe ad occupare se digitalizzata oltre al tempo che si perderebbe nell'operazione pratica di scansione e successivamente di organizzazione del materiale digitale così prodotto. E' ben vero che la trasformazione da carta a media digitale implica necessariamente una riduzione dello spazio occupato dal materiale, tuttavia non sempre questa operazione è consigliata. Non va dimenticato infatti il fascino che esercita il fascicolo cartaceo e la sua comprenetazione d'uso quotidiano, come la lettura a letto o in treno: attività difficilmente sostituibili con un PC, non fosse altro che per i risvolti psicologici che ci fanno percepire certe attività come rilassanti.

Venendo al calcolo dello spazio occupato facciamo un esempio pratico: supponiamo di procedere con la scansione di una rivista a colori di cento pagine. Scegliamo il formato A4 e una impostazione di parametri buona ma non eccelsa (tipo 300x300 a 16.000 colori).

Le cento pagine ci occuperebbero almeno un'ora del nostro tempo e diciamo (ma ad essere bravi) di aggiungerci un'ora ulteriore per organizzare le scansioni, produrre magari in PDF, immagazzinarlo e catalogarlo, etc... Come spazio a disco si viaggia sui 30 mega per mantenere una qualità discreta che non faccia rimpiangere troppo l'originale.

Lavorando alacremente come formiche, una collezione di riviste (diciamo un centinaio di numeri) impiegherebbe un mese del nostro tempo, a lavorarci otto ore al giorno, e tre giga di disco, diciamo un DVD. Indubbiamente uno o anche tre supporti ottici, contenitore compreso, occupano uno spazio fisico irrisorio rispetto a cento numeri cartacei di una rivista di un centi-

naio di pagine, non si discute! Quello che spaventa di più è comunque il tempo necessario per l'operazione. Qui vorrei spezzare una lancia a favore di questa attività che varrebbe al pena organizzare in una sorta di consorzio collaborativo. Cioè se ogni appassionato di retro-computer si impegnasse a digitare 50 riviste in un anno, ovviamente condividendo il lavoro con gli altri, cento persone potrebbero in un anno portare a casa 50.000 pagine di riviste, l'equivalente di 500 riviste. Probabilmente nel giro di tre-cinque anni potremmo disporre di tutte le riviste edite in Italia dal '80 al '90, cioè il pacchetto più interessante in assoluto!

A parte questi "sogni" che oltre a garantire una ulteriore chance nella conservazione del materiale, permetterebbero una maggiore diffusione della conoscenza, vediamo gli aspetti di conservazione del materiale cartaceo in sè e per sè. In questo campo le conoscenze biblioteconomiche sono molto avanzate: in fondo sono secoli che tentiamo di preservare libri di pergamena prima e di carta poi, per non citare i papiri egiziani...

Materiale e ambiente

Ciò che influenza in maniera determinante la conservazione della carta è la qualità del materiale e l'ambiente in cui esso viene conservato.

La carta deve essere di buona qualità, gli inchiostri altrettanto. Questa è la base, poi sta a noi cercare il posto più adatto fra quelli a nostra disposizione. Bisogna scartare le cantine (troppo umide) e il bagno (per lo stesso motivo), ma anche le soffitte non sono proprio l'ideale, sopratutto se soggette a variazioni notevoli della temperatura estate/inverno. Molto meglio un ambiente anche non troppo ideale ma che abbia una limitata escursione termica piuttosto che sottoporre il materiale cartaceo a notevoli sbalzi di temperatura.

I parametri ideali, cui i bibliotecari e i conservatori museali tentano di adeguarsi, sono in sostanza: umidità relativa del 35% e temperatura attorno ai 22 gradi. Il pericolo maggiore è comunque quello dell'umidità, tanto che nelle conservazioni professionali vengono prelevati periodicamente campioni di materiale per misurarne il livello e per controllare lo stato generale di conservazione.

La luce

La luce è un'altra fonte di pericolo: alcune carte e soprattutto gli inchiostri sono sensibili alla luce ultravioletta. Niente Sole diretto quindi e possibilmente nemmeno troppa luce, la classica luce diffusa delle sale di lettura con pareti di legno assorbenti sarebbero l'ideale...

Non potendo fare di meglio esistono dei filtri UV abbastanza efficaci da applicare ai vetri delle finestre. Non dovrebbero entrare nelle nostre liste ma le carte più in pericolo

sono quelle dipinte con acquarelli, i manoscritti e in generale le foto esposte sulle pareti. Non sarà un segreto per nessuno credo, ma musei ed esposizioni varie usano esporre copie e non originali, non tanto per evitare i furti, ma proprio per evitare il degrado delle opere originali.

I giornali hanno anche il problema dell'ingiallimento della carta, presente anche nelle riviste e nei libri, ma in quest'ultimi solitamente molto più lento. L'ingiallimento deriva principalmente dall'esposizione alla luce che altera la composizione chimica superficiale della carta riportandola ad una tinta più naturale (il bianco è ottenuto con addittivi chimici).

Inquinamento e polvere

L'inquinamento degli ambienti, principalmente il fumo, determinano un degrado mlto veloce del materiale. La carta assorbe in maniera massiva e purtroppo praticamente permanente il fumo determinando anche uno sgradevole odore del media stesso che potrebbe generare repulsione ma arrivare fino alla nausea da parte dei malcapitati soggetti che desiderano sfogliarne le pagine. Ovviamente anche tutti gli inquinanti acidi (ad esempio il liquido che potrebbe fuoriuscire dalle pile alcaline esauste dimenticate su uno scaffale o peggio sopra una pila di riviste, sono micidiali per il materiale.

La polvere è un'altro nemico che potrebbe rendere inutilizzabile parte dell'oggetto, soprattutto se si tenta di rimuoverla in maniera non adeguata. L'effetto principale è una sorta di "fango" superficiale che si forma dalla combinazione degli strati di polvere con l'umidità comunque presente.

Non citiamo per dignità la presenza di topi e altri roditori che in mancanza di altro fanno volentieri uno spuntino con le nostre riviste preferite...

Attenzione anche ai parassiti: ci sono tarme e una lista di altri nemici dei libri che ne rosicchiano le pagine e riescono a bucare parte a parte interi volumi di grosse dimensioni. Per questi sgraditi ospiti ci sono in vendita degli insetticidi specifici ma si può usare preventivamente anche la naftlanila o per i naturisti dei mazzetti di malva essiccata.

Immagazzinamento

I professionisti consigliano uno stoccaggio delle riviste organizzato in grosse buste di materiale alcalino (che neutralizza gli effetti degli inquinanti acidi) da mettere sugli scaffali appoggiati in piano piuttosto che in piedi. Bisogna ricordare che il pericolo acido viene dalle riviste stesse in quanto gli inchiostri hanno un PH superiore al grado 7. Un'altro errore che si compie in buona fede è quello di non togliere il "naylon" nel quale vengono confezionate le riviste ad esempio le la spedizione via posta. Questa

protezione è eccezzionale come anti-umidità e anti-polvere, ma può caricarsi elettrostaticamente e mettere in funzione delle indesiderabili reazioni chimiche sugli inchiostri che ne vengono in contatto.

manipolazione del materiale

Pulirsi le mani prima di toccare il materiale cartaceo e possibilmente evitare di mangiare quando si stanno sfogliano libri e riviste. Assolutamente deleterio è il deposito oleoso che potrebbe trasferirsi dalle nostre mani alla carta: è molto difficile da togliere.

Preferire l'uso di una matita a punta morbida quando si lavora su materiale cartaceo. Se proprio non potete fare a meno di fare annotazioni e sottolineature, almeno fatele con la matita!

Una perfetta conservazione deve prevedere una certa cura nel trattamento del supporto fisico che dovrebbe risultare il più possibile integro. Ad esempio curare che non ci siano piegature e le classiche "orecchie" sulle pagine ed evitare di mettere del materiale pesante sopra le riviste in modo da evitare che possa trasferire la propria impronta ai sottostanti fascicoli. Ovviamente, ma questo dovrebbe essere banale, non appoggiare bicchieri di liquido, nemmeno di semplice acqua, sopra le riviste e i libri.

In caso di malcapitata bagnatura del fascicolo o anche di una sola pagina è necessario mettere in atto tutta una serie di accorgimenti per evitare che il danno diventi irreparabile: asciugare il grosso con carta assorbente o anche con carta igienica se non disponibile altro e stendere le pagine o l'intero fascicolo con le pagine ben staccate.

Non stendere le pagine al sole diretto ed evitare l'azione di asciugatura trascinando l'assorbente in giro per la pagina: semplicemente metterlo sopra come un tampone e sollevarlo senza strisciare. Ricordare che se si usa carta igienica, questa si scioglie in acqua e quindi non può essere lasciata sulla "ferita", ma asportata appena si è impregnata abbastanza del liquido.

In alcuni casi è necessario diluire il liquido anche se questo comporta una bagnatura più estesa e profonda del foglio. E' il caso di inchiostro (ad esempio quello per le stampanti ink-jet) malauguratamente versato.

Se poi non si può fare proprio nulla, almeno cercare di arginare il danno e salvare le altre pagine/parti del documento.

Dopo l'asciugatura se è il caso si possono stirare le pagine con un ferro da stiro, ovviamente regolato a bassa temperatura e mai direttamente messo a contatto con il foglio. Evitare la stiratura a vapore!

Qualche volta vanno prese delle drastiche decisioni. Ad esempio se vi si è bagnato un fascicolo intero e non solo tre pagine, conviene smembrare del tutto il fascicolo ottenendo dei fogli singoli che possono asciugare meglio per evitare che due pagine si incollino fra loro. Il fascicolo può poi essere rilegato e la rivista conserva in pieno il proprio valore.

*[traduzione a cura di **Tn**]*

Bibliografia

Preserving Works on Paper: Manuscripts, Drawings, Prints, Posters, Maps, Documents

http://www.digitalpreservation.gov/you/digitalmemories.html

Come eravamo...

La storia dei sistemi e degli uomini che hanno creato un mondo nuovo.

Homebrew Computer Club



L'autore di questa memoria, Bob Lash alle prese con una presentazione di alcune foto d'epoca, in una rassegna dedicata alla storia del Club.

Ricordo di un membro del Homebrew Computer Club

By Bob LASH

[bob@bambi.net]

Traduzione e adattamento di [Tn]

Il primo incontro con un computer l'ebbi all'età di sei anni, esattamente nel 1963, quando partecipai ad un esperimento sull'apprendimento della matematica, condotto dal dott. Patrick Supper. Mi misero davanti ad un piccolo schermo, come quelli che ci sono ora nei video citofoni, e mi chiesero di pigiare alcuni tasti in risposta alle figure che vi apparivano.

Dopo l'esperimento mi mostrarono la stanza attigua dove c'era il computer che raccoglieva ed elaborava le risposte del test. Ricordo un grande armadio con luci bianche che lampeggiavano e un signore alto che mi spiegava che quella macchina si chiamava DEC PDP-1. Invitato a premere un tasto vidi due bobine che si mettevano in moto al mio comando: avevo fatto (senza avere nessuna idea di cosa significasse) il mio primo Dec-Tape; una emozione che non ho mai più dimenticato!

A 10 anni iniziai a costruire ogni tipo di circuito elettronico fra i quali anche alcuni pannelli pieni di flip-flop realizzando le saldature assieme a mio cugino Arnold Gold.

Questi servivano a suo padre per realizzare un "albero di luce" per l'esposizione di San Francisco. I

visitatori erano invitati a battere le mani o a parlare e l'albero si illuminava dalla base fino alla cima in relazione all'intensità dei suoni.

Mi piaceva anche costruire piccoli razzi (eravamo all'epoca dell'Apollo); per questo ero entrato in contatto con i sistemi elettronici della Micro Instrumentation and Telemetry Systems (MITS, vi ricorda qualche cosa?). Non era ancora l'Altair, ma le luci lampeggianti c'erano tutte e mi affascinava poterle comandare a mio piacimento.

Nel 1973 ero uno studente della scuola di Palo Alto, PAUSD (Palo Alto District School), dove diventai operatore del sistema HP2000F, sotto la direzione del dott. Bruce Keepes; un calcolatore programmabile in BASIC con un sistema di time-sharing scalabile fino a 32 utenti. Successivamente il sistema venne upgradato ad un HP3000 che con Bruce Keepes e Ian Yoch lo collegammo attraverso un modem a 300 boud. Ricordo ancora il pomeriggio che riusci a collegare l'HP3000 con il suo predecessore HP2000F attraverso un accoppiatore acustico e conversare via Teletype con l'operatore dello stesso, un certo Greg Dolkas, che poi andò a lavorare proprio per la Hewlett-Packard.

Il People's Computer Center a Palo Alto era davvero un bel posto per gli hobbisti dell'epoca. Possedeva un PDP-8/E e un PDP-8/L con quattro teletype ASR33 collegate: un modo eccellente per interagire con i sistemi, anche se riservata a "veri uomini" (vedi figura 2).

Il primo annuncio pubblico di una riunione fra noi studenti appassionati di computer, venne appesa sulla bacheca della sala PALY, che era la sala della scuola di matematica della High School di Palo Alto, dove trovavano posto sei rumorosissime telescriventi di tipo KSR-33.

Il meeting era organizzato nel garage di Gordon French: data e luogo dell'incontro era scritto sul retro di una busta usata.

Ci trovammo quindi a casa di Gordon; eravamo io, Mike Fremont e Ralph Campbell. Eravamo tre giovincelli timidi (io lo sono tutt'ora), ma rimanemmo assolutamente a bocca a aperta quando Gordon ci mostrò il suo sistema 8008 che usava una scheda da 16 Kbyte basata su chip di shift register come memoria di massa. Gli shift regi-

Eccomi, giovanissimo (ma non sono cambiato poi molto, per fortuna), nella mitica stanza dei bottoni presso lo SLAC.





Il Club faceva uscire una fanzine periodica, questa è la copertina di uno dei primi numeri.

ster erano organizzati come una anello e il sistema era programmato in modo che la lettura/scrittura su un registro veniva eseguita quando l'indirizzo sull'anello era quello corretto. Una cosa davvero geniale che ci lasciò stupiti.

Era stato costituito di fatto l'Hombrew Computer Club.

All'homebrew computer club l'eccitazione era elettrica e durante il periodo dal '74 al '76 costrui un minicomputer programmabile a 12 bit usando un centinaio circa di integrati TTL. Ce l'ho ancora in garage, anche grazie alla comprensione di mia moglie...

Mio cugino Arnold Gold ha partecipato alla costruzione sedendo con me nel garage gelido per ore alla notte, inpegnato a wrappare i componenti. Il computer era tutto cablato con file wrap-out, una tecnica molto di moda al tempo che era anche più veloce da realizzare e soprattutto da correggere rispetto ad un circuito stampato. Non potevo disporre di un microprocessore perché usare un 4004 o addirittura un chip 8008 avrebbe comportato un costo proibitivo.

Mi pagavo i componenti lavorando presso lo spaccio di bibite "Da Edy", vicino al Campus di Stanford. Una notte ritornando a casa dopo il giorno di paga con un chip di memoria da 256 bit (non byte, bit!), mia madre mi apostrofò duramente: -"E questa sarebbe la tua paga di un mese?".

Sul pannello frontale ci sono quasi cento lampadine a incandescenza e 56 switch che ero riuscito a recuperare smontando tre telescriventi guaste datemi da un amico (David Bell).

Recentemente ho ricontattato Gordon French grazie alla Rete e mi ha stupito molto apprendere che egli è ancora segretario dell'Homebrew Computer Club! Egli è attualmente impegnato nella scansione dei vecchi bollettini del club. La seconda riunione del Homebrew si è svolta in una vecchia scuola, e Steve Dompier ha portato nel suo Altair, e ha messo una radio a transistor sopra di esso. In mezzo allo stupore generale ha fatto suonare la canzone "Fool on the Hill" lavorando su un programma dell'Altair.

Poco tempo dopo ho usato anch'io la sua idea ed ho collegato un piccolo altoparlante all'uscita open-collector che pilotava una della lampade ad incandescenza della mia macchina a registri. E' stato divertente provare a riprodurre canzoni lavorando sulla temporizzazione dei programmi.

Ralph Campbell aveva costruito una delle prime Altair e posto una scritta "OBSOLETE" sul pannello frontale. Stanco di lavorare a suon di switch, egli assemblò la sua scheda CPU con a bordo una

EPROM con il monitor e l'uscita per un video CRT. Ralph più tardi ha contribuito alla realizzazione delle architetture RISC I e II presso Berkeley e successivamente diventò uno degli sviluppatori dello UNIX Berkeley (fino alla versione 4.4).

Un'altro membro diventato poi famoso è John Draper (alias "Captain Crunch") che partecipava regolamente alle riunioni fin dal tempo in cui lavorava per la Call Computers.

Nel 1976 Steve Wozniak mi passò una copia delle sue routine matematiche per trattare i calcoli in virgola mobile con il micro 6502. Ho passato una intera notte a digitare byte-per-byte per trasferirlo su un nastro perforato. Avevo appena aggiunto un 6502 al mio sistema e vedere il tutto funzionare è stata una emozione indescrivibile!

Spergel Marty (M&R Elettronica) aveva il talento di poter recuperare parti elettroniche per i membri dell'Homebrew e partecipò alla organizzazione di un gruppo di acquisto. Ricordo che era veramente un bel ragazzo.

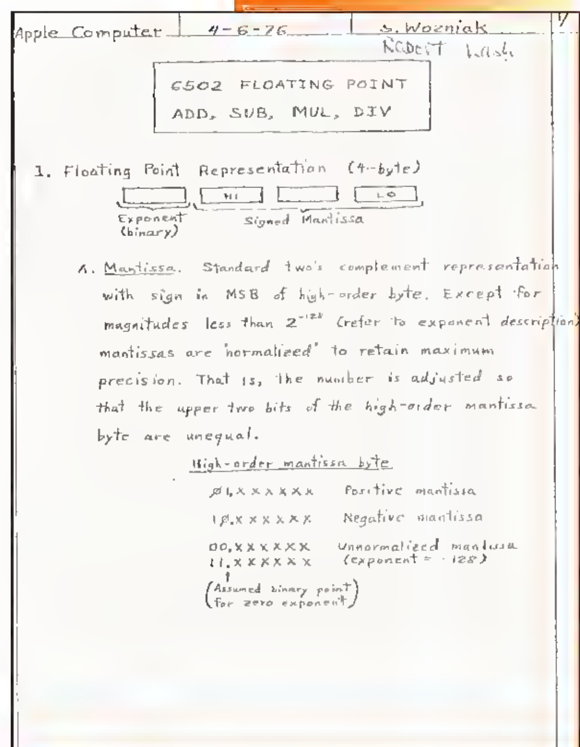
Gordon French era sempre molto interessato al lavoro di ciascuno. Il ricordo che ho di lui all'epoca era di uno "spirito-guida", una sorta di mentore. La nostra differenza di età era grande e i suoi capelli brizzolati contribuivano a questa mia percezione. Dal punto di vista della realizzazione dei sistemi potevamo però definirci coetanei.

Lee Felsenstein, che progettò il Pennywhistle Modem e successivamente sviluppato il SOL computer (insieme a Bob Marsh) della Processor Technology, lo portò ad una riunione del club. Il club era cresciuto al punto che dovemmo utilizzare l'auditorium allo SLAC (Stanford Linear Accelerator Center). Mio cugino Arnold Gold acquistò un SOL, così che ebbi l'opportunità di scrivere il codice per esso: un programma per la gestione di una coltivazione (legale) di funghi.

Era veramente una ottima macchina e sono convinto che se solo avessero avuto i capitali per progettare un cabinet plastico al posto di usare un contenitore di legno e metallo di un amplificatore stereo, la Processor Technology sarebbe diventata un'altra Apple Computer.

L'unico membro dell'Homebrew Computer Club con il quale sono rimasto in contatto stretto fino ad ora è Mike Fremont. Scrisse il programma MemoryMate per la Broderbund e abbiamo lavorato assieme alla realizzazione di WebChat, una dei miei lavori dell'epoca pionieristica del Web.

Assieme a Mike facevamo delle escursioni al SAIL



Apple Computer | 4-6-76 | S. Wozniak | 7

6502 FLOATING POINT
ADD, SUB, MUL, DIV

1. Floating Point Representation (4-byte)

[ ] [HI] [ ] [LO]

Exponent (binary) — Signed Mantissa

A. Mantissa. Standard two's complement representation with sign in MSB of high-order byte. Except for magnitudes less than 2^{-128} (refer to exponent description) mantissas are 'normalized' to retain maximum precision. That is, the number is adjusted so that the upper two bits of the high-order mantissa byte are unequal.

High-order mantissa byte

Ø1.XXXXXX Positive mantissa
1Ø.XXXXXX Negative mantissa
00.XXXXXX Unnormalized mantissa
11.XXXXXX (exponent = -128)

(Assumed binary point for zero exponent)

La routine per la gestione dei numeri floating point nell'Applesoft, scritta di suo pugno direttamente da Steve Wozniak.



Il nostro DEC PDP/11, ora funzionante nelle mie stanze. Una delle poche cose rimaste dal mio primo tentativo di mettermi in proprio.

(Stanford AI Lab) dove trovavamo sempre qualcuno intento a giocare a "Lunar Lander" su un grande schermo grafico o passavamo ore a guardare un braccio robotizzato che cercava di impilare dei cubi di legno o comporre delle figure con gli stessi.

Una notte un orgoglioso studente ci mostrò un rack che conteneva un milione di parole di memoria per il loro PDP-10. Rimanemmo allibiti e travolti dall'enormità della realizzazione.

Al laboratorio di Stanford si è svolto anche il nostro primo incontro con ARPANET. Mike e io con una telescrivente a carta termica della HP (era il modello TI Silent 700, è chiaro il perché di quel "Silent": avete mai provato a lavorare in una stanza con altre cinque o peggio dieci persone intente a battere comandi su una teletype meccanica?). Il laboratorio SAIL organizzava degli eventi chiamati "Explorer Scout" e in quelle occasioni gli studenti potevano, in piccoli gruppi, arrivare, come facemmo noi due, a giocare a scacchi con un computer installato a Londra! A quei tempi erano poche le persone che potevano "andare in rete": giusto gli uffici militari e le università avevano un collegamento. Certo ne è passata di acqua sotto i ponti!

Andavamo anche frequentemente a Tressidor dove c'era una specie di club studentesco per giocare a Space War, mangiare patatine e bere gazzosa.

Ricordo la fregatura che prendemmo Mike e il sottoscritto quando leggemmo l'annuncio della vendita di materiale elettronico a prezzi stracciati. L'annuncio, apparso sul numero di settembre di Elettronic News del 1974, chiedeva il pagamento anticipato della merce, Mike ordinò un 8080 e io dei chip di memoria statica 2602B da 1 Kbx1. Il microprocessore non arrivò mai e i chip di memoria erano sbagliati! La sigla era stata stampata sopra la precedente ma era palese l'imbroglio: nei chip mancavano due pin! Quanto ho maledetto quel truffato-

re! Fra l'altro non è che si navigava nell'oro e tutto il materiale elettronico che riuscivamo a mettere assieme si può ben dire che fosse frutto del nostro sudore!

Entrai a Berkeley nel 1975 così che potei frequentare l'homebrew molto meno spesso di prima e cominciava ad essere difficile spendere intere notti a scrivere codice o lavorare a progetti di ampliamento del mio sistema autocostruito. Così il tempo che trascorsi presso la scuola medica dell'UCSD mi obbligò a perdere il contatto con quella parte della mia vita che considero ancora oggi la più preziosa! Non ebbi l'opportunità di rivedere l'amico Gordon French dal 1979 circa fino al 1999 quando ricevetti la richiesta da parte della rivista US Magazine, che Gordon aveva messo sulle mie tracce, per scrivere un articolo sui giovani hakers dell'epoca dei primi rudimentali home costruiti in casa. Poi dell'articolo non se ne fece nulla perché in realtà quello che volevano erano giovani hackers sotto i 21 anni, non uno come me, ormai "anzianotto".

Nel 1982 scrissi un interprete LOGO per i computer Timex e Sinclair, programma commercializzato con il nome di TINY LOGO da una ditta canadese.

Pensavo veramente che un sistema che costava solo 49 dollari potesse diventare un prodotto di massa ed attirare nel campo della programmazione moltissime persone, ma mi sbagliavo: l'utilizzatore medio non era affatto interessato alla programmazione, preferiva comprare programmi già fatti ed usar ela macchina con questi piuttosto che costruirsi lui stesso gli strumenti che gli servivano.

Avevo preso l'ispirazione dall'interprete TINY BASIC scitto da Tom Pittman per l'homebrew computer club dimostrando fra l'altro che si poteva scrivere un interprete nei pochi K di memoria disponibili.

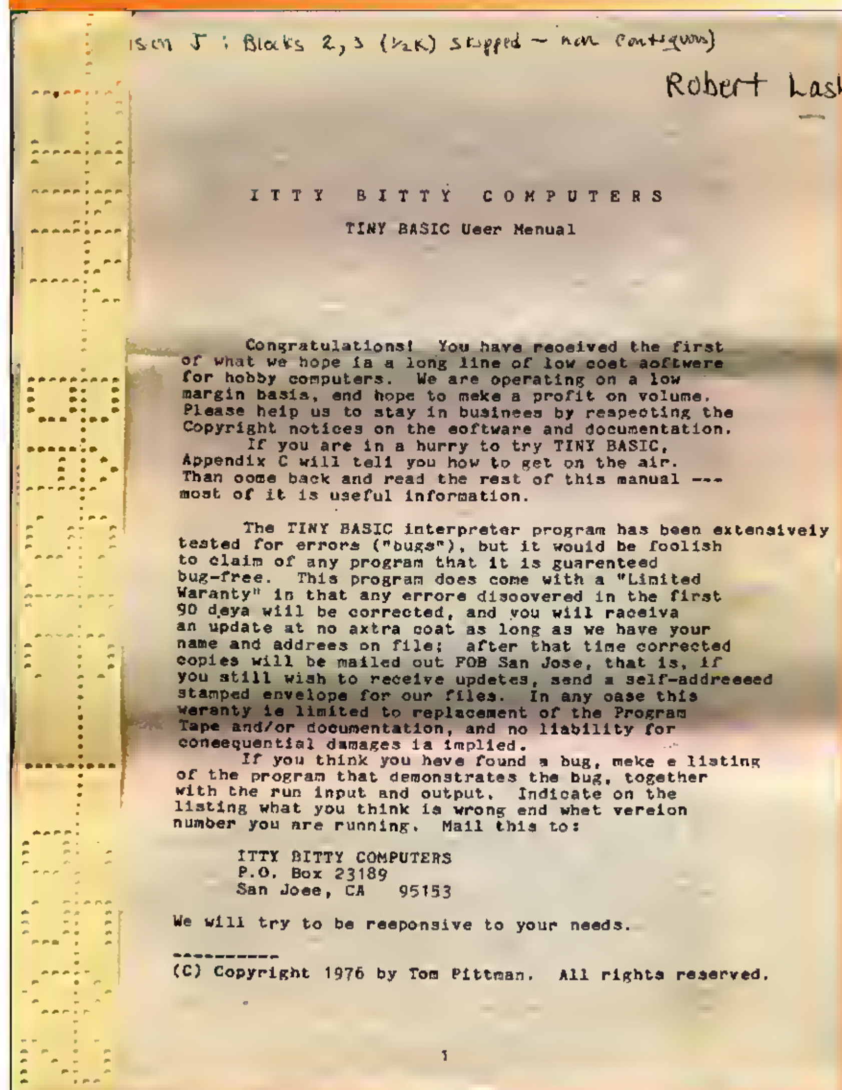
Dopo la scuola di medicina, nel 1984 ebbi l'idea di fondare una società, assieme ad un giovane collega che era anche ingegnere, per costruire apparecchiature medicali comandate da computer. Il nostro campo di specializzazione si orientava alla costruzione di apparecchiature per la laparoscopia e la chirurgia generale ed estetica. Nel 1989 decisi di dare vita ad uno spin-off diventando il presidente di una nuova società che avrebbe dovuto sfruttare industrialmente un nostro brevetto .



Il pannello frontale del PDP 8/I da me restaurato e ora funzionante.



Il Tiny Logo da me progettato e scritto. Sotto il manuale del Tiny Basic, al quale devo l'ispirazione.



I T T Y   B I T T Y   C O M P U T E R S

TINY BASIC User Manual

Congratulations! You have received the first of what we hope is a long line of low cost software for hobby computers. We are operating on a low margin basis, and hope to make a profit on volume. Please help us to stay in business by respecting the Copyright notices on the software and documentation.

If you are in a hurry to try TINY BASIC, Appendix C will tell you how to get on the air. Then come back and read the rest of this manual --- most of it is useful information.

The TINY BASIC interpreter program has been extensively tested for errors ("bugs"), but it would be foolish to claim of any program that it is guarenteed bug-free. This program does come with a "Limited Waranty" in that any errors discovered in the first 90 days will be corrected, and you will receive an update at no extra cost as long as we have your name and address on file; after that time corrected copies will be mailed out FOB San Jose, that is, if you still wish to receive updates, send a self-addressed stamped envelope for our files. In any case this waranty is limited to replacement of the Program Tape and/or documentation, and no liability for consequential damages is implied.

If you think you have found a bug, make a listing of the program that demonstrates the bug, together with the run input and output. Indicate on the listing what you think is wrong and what version number you are running. Mail this to:

ITTY BITTY COMPUTERS
P.O. Box 23189
San Jose, CA 95153

We will try to be responsive to your needs.

(C) Copyright 1976 by Tom Pittman. All rights reserved.

1

Un'altra idea che ci parve potesse avere successo era la messa a punto di un dispositivo per il controllo anticoncezionale basato sulla misurazione della temperatura basale. Avevamo il brevetto e con questo unico obiettivo decidemmo di fare uno spin-off e creare una società ad-hoc della quale io assunsi la direzione. Trattandosi di un dispositivo medico destinato al pubblico, erano necessari degli step che sottovalutai: una lunga fase di studio preventivo, fatta da istituti specializzati, che dovevano dichiarare il prodotto non dannoso per la salute e un test diffuso con centinaia di utenti. Ci mancò il capitale per finanziare quest'ultima fase, nonostante i nostri primi test dimostrassero la bontà dell'idea.

Di questa esperienza imprenditoriale in prima persona mi è rimasto il PDP-11/23 Plus, che era la macchina di sviluppo della società. Funziona ancora a casa mia! Sono veramente un fanatico di questo mini con sistema operativo RT11 e assembler MACRO-11. Faccio parte della PDP Unix Preservation Society e ti invito a partecipare se anche tu ami questa leggendaria macchina.

Dopo la prima esperienza post-laurea, andai a lavorare per Mike Fremont nella sua ditta che si chiamava allora "Fun University Network". Si capisce dal nome quale era il campo applicativo della società: software di comunicazione

ma soprattutto front-end per la comunicazione via rete e i servizi annessi. Questa società cambiò così tanti nomi per adattarlo meglio all'evoluzione dell'informatica e degli scopi della ditta, che mi piace ricordarli qui nell'ordine in cui furono ad essa assegnati nel tempo:

Learning Things, Gecko Learning Products, Brainstorm Software, Fun University Network (FUN), Springboard, Internet Roundtable Society, WebChat Communications, e infine WebChat Broadcasting System (WBS).

Per tutto il tempo lavorai allo sviluppo di applicazioni e servizi on-line che si caratterizzarono nel tempo per l'approccio innovativo.

All'epoca in cui fu acquisita da Infoseek e conseguentemente integrata in GO.com (di proprietà della Disney), la WBS aveva costruito la più grande comunità on-line esistente al mondo.

Dopo la vendita Mike fondò la Zorch con lo scopo di sviluppare siti web per giochi interattivi on-line. Lo seguì anche in questa avventura ma due anni più tardi abbandonai questa società per entrare nella Driving Media, una società ideata da David Morse, uno dei fondatori della Amiga Computer.

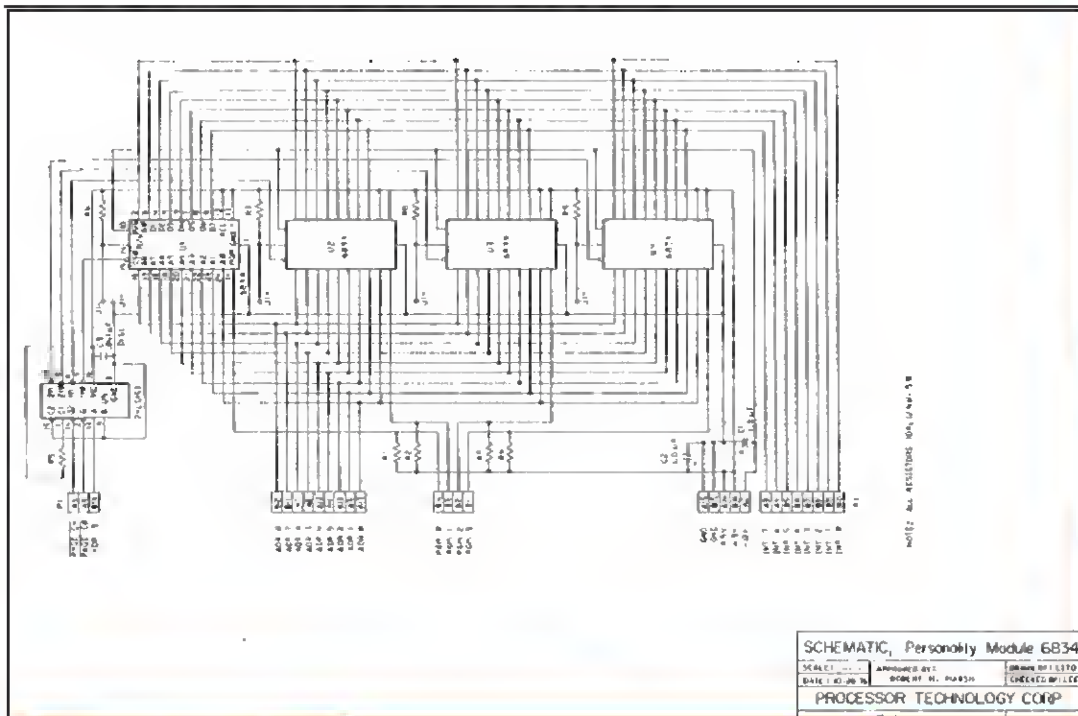
La mia carriera era arrivata all'apice e assunsi il ruolo di manager della ricerca e sviluppo, lo stesso ruolo che ora ho alla VIOptix, la ditta per la quale lavoro attualmente.

Un restauro al quale tengo molto è quello del PDP-8/I che acquistai assieme a Mike Fremont nel 1999. La macchina non funzionava, anzi, le sue condizioni richiesero la sostituzione di moltissime parti, compreso l'intero cablaggio. Comunque con l'aiuto di Mike riuscì nell'impresa e ora le 22 lampade ad incandescenza (tutte guaste all'arrivo nel mio laboratorio) lampeggiano

Un'altra delle mie realizzazioni: il compilatore per il linguaggio FABOL, da me ideato.

1/19/76

```
3   print ""
    print "fabol compiler"
5   print "pass 1"
    print "enter source tape"
/ initialize pass flag
    c(63)=1
/ initialize text line ctr
    c(1)=0
/ initialize abs adr ctr; in pass 2 abs ctr saved in c(62)
    c(2)=0
/ initialize label table ptr, subr table ptr
    c(3)=1
    1(0)-1
    a(1)="input,print,stop,goto,if,subrtn,•,return,end, ."
/ init subrtn return ctr
    c(59)=0
10 *inputs
```



Uno schema dell'Atari ST originale, fatto circolare al club dal suo progettista. Il progetto definitivo fu cambiato ma qualcosa delle idee originali rimasero.

"felici" sul pannello frontale.

Il sistema ha 8K di CORE memory e la sua capacità di calcolo si aggira attorno al MIPS. Ho realizzato anche un collegamento con una console Linux che mi permette di controllare la macchina anche da Internet.

Come il mitico FOCAL del 1969, il mio PDP-8/I è in grado di far girare la versione 4K di Chess.

L'Homebrew Computer Club non esiste più dal 1986, ma il suo spirito vive ancora. Io vedo la stessa eccitazione di allora negli occhi dei giovani che si avvicinano al mondo di Internet e del computer, mi identifico con la loro immaginazione e sò che stanno guardando nel futuro.

Una rimpratriata l'abbiamo organizzata il cinque marzo 2001 presso lo SLAC, grazie all'interessamento e al supporto dello Stanford-Palo Alto Macintosh User's Group, alla quale hanno partecipato molti membri fondatori e altri che ne hanno avuto parte attiva, come il sottoscritto.

In generale tutti i membri attivi dell'HCC hanno fatto una carriera brillante e per la maggior parte sono ancora nel settore dell'innovazione tecnologica.

Len Shustek ha fondato la Network Associates ed è ora direttore del Computer Museum History Center in California.

John Draper (il mitico Capitan Crunch) non ha certo bisogno di presentazioni. Oggi lavora su sistemi Firewall.

Ralph Cambell è alla SUN; Dan Sokol lavora nel settore della microelettronica e noi tutti gli siamo grati per la grande quantità di informazioni che ci passava all'epoca del club.

Questa è la mia personale storia dell'Homebrew Computer Club, vissuta con la nostalgia che non posso nascondere ma con la consapevolezza, come ho già peraltro affermato, che ciò che noi eravamo allora esiste ancora e forse in

maniera più grande, sicuramente in maniera più diffusa grazie alla rete.

Dispiace solamente che per il 99% e forse più delle persone che usano il computer, i primi faticosi tentativi messi assieme da pochi "pazzi illusi" nelle loro stanzette di adolescenti o in improbonibili garage trasformati in laboratori, non dicano sostanzialmente nulla, anzi ne siano completamente all'oscuro.



Io alle prese con il progetto Restoring PDP1 presso il Computer History Museum.

Dimenticare il passato o peggio non sapere quale esso sia stato, la storia insegna che è pratica pericolosa oltre che impoverimento culturale.

[a cura di Tn]

Nella foto ecco gli "Homebrew Originals": da sinistra a destra in prima fila Len Shustek, Gordon French, Marty Spergel, Bob Lash (io), Ralph Campbell, Mike Carlisle, Walter Bryant, George Oetzel, Harry Garland, Allen Baum, Lee Felsenstein, Dennis Allison (con la testa girata), Gene Wallace. Dietro: Roger Melen, Bob Marsh, Fred Balin (qualcuno è nascosto quasi del tutto).



Biblioteca

Le monografie vecchie e nuove che rappresentano una preziosa risorsa per chi ama il mondo dei computer in generale.

Scheda

Titolo:

AIM 65 - Laboratory Manual and Study Guide

Autore:

Leo J. Scanlon

Editore:

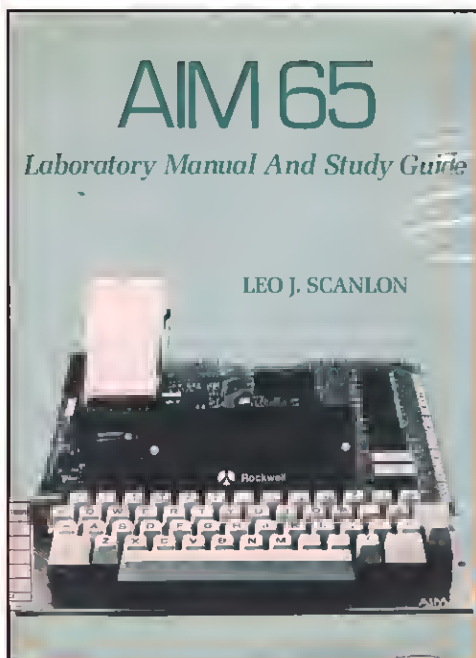
John Wiley & Sons

Anno: 1981

Lingua: Inglese

ISBN: 0 471 06488 2

AIM 65 - Laboratory Manual and Study Guide



Nel 1981 occuparsi di micro calcolatori significava confrontarsi con logiche digitali e programmazione a basso livello. Kit come l'AIM65 della Rockwell International avevano lo scopo di costiuire un laboratorio a supporto della teoria dei microprocessori.

Con questo spirito si presenta questo volume, scritto da un dipendente della Rockwell, produttrice del kit.

Bisogna dire che la piastra della Rockwell, equipaggiata con un mp6502, poteva dirsi l'archetipo di questo genere di prodotti: vera tastiera alfanumerica, ampio display di 40 caratteri (una sola riga), stampante integrata e slot per espansione. Tutto quello che serve insomma per condurre esperimenti e realizzare prototipi di controlli o semplice programmazione.

Il testo, di grande formato, consta di 180 pagine e si configura come un testo di supporto per un corso di laboratorio di micro elettronica. A fianco di una breve introduzione teorica ai singoli argomenti, compresa la descrizione dell'hardware, trovano posto in ogni capitolo una serie di esperimenti da condurre sul sistema. Alla fine del testo le soluzioni complete e commentate, a supporto di una maggiore comprensione.

Gli esperimenti sono progressivi per difficoltà; si comincia con semplici programmini in codice macchina per passare poi alla costruzione di routines più complesse, come l'implementazione delle operazioni aritmetiche di moltiplicazione e divisione (che i micro a 8 bit non eseguivano in microcodice, a differenza di come avviene oggi).

Infine vengono affrontati argomenti decisamente più impegnativi: l'I/O, con il supporto del chip VIA embedded nella piastra, e la gestione degli interrupt.

La parte finale del testo spiega come utilizzare il monitor contenuto

in ROM per riutilizzare le routines, oltre che per controllare il sistema. Dulcis in fundo l'Assembler, accoppiato con un elementare Text Editor, sempre presente nella ROM, che apre la strada ad un approccio più ordinato e professionale verso la programmazione dei micro sistemi.

Conclusione.

Un testo che decisamente non può mancare per chi possiede questo kit. E' vero che esistono numerosi libri che trattano del 6502, tirando in ballo anche l'hardware AIM65 e che la stessa documentazione ufficiale del progetto è molto completa, ma se si vuole capire bene cosa significasse avvicinarsi alla programmazione dei micro calcolatori se ne ha qui un esempio pratico.

[Sn]

101

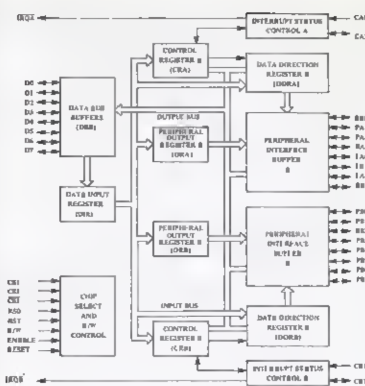


Figure 13-1. Block Diagram of the R6520 PIA
(Courtesy of Rockwell International)

- $AC00 is used to access Peripheral Interface Buffer A (called Data Register A in the AIM 65 User's Guide) and Data Direction Register A
- $AC01 is used to access Control Register A
- $AC02 is used to access Peripheral Interface Buffer B (called Data Register B in the AIM 65 User's Guide) and Data Direction Register B
- $AC03 is used to access Control Register B

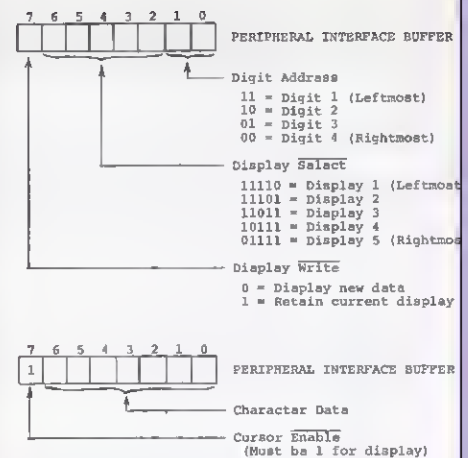


Figure 13-2. Peripheral Interface Buffers for Display PI

110

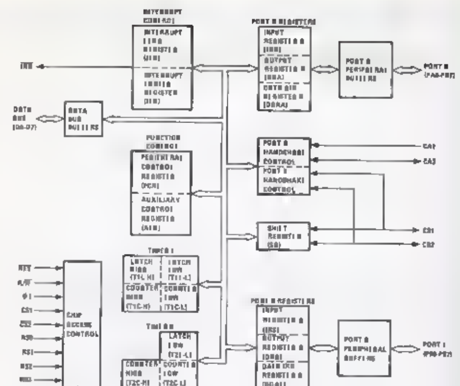


Figure 14-1. Block Diagram of the R6522 VIA
(Courtesy of Rockwell International)

way of knowing when the Reset button will be pressed, the program must be capable of monitoring the elapsed time continuously, at 10-millisecond intervals, until someone presses Reset.

As you know from reading Section 8.7 of the AIM 65 User's Guide, Timers 1 and 2 both have "one-shot" timing modes, which will produce a specified time interval one time. These modes could certainly be employed to solve our problem, but that would be inefficient, since the timer would have to be re-started after each 10-millisecond interval. A much

BBS

A colloquio con i lettori

Posta

Da Antonio

Carissimi,

è stata una gioia constatare che la pausa estiva nelle uscite della rivista non ha determinato quello che temevo: l'abbandono dell'iniziativa. Continuate, vi prego, la rivista mi piace molto ed è diventata un appuntamento fisso per me, come l'aprire il baule dei ricordi in soffitta. Volevo dire che la mi piace leggerla, mi fa tornare se non bambino certo un sacco più giovane, ma proprio un sacco!

Penso sempre anche ad un'altra cosa, cioè a tutte le scelte sbagliate che ho fatto nella vita, tecnologicamente parlando. Ad esempio a non aver capito che Windows avrebbe scalzato il DOS o che l'amato Amiga sarebbe durato tanto poco...

Certo con il senno di poi è facile darsi delle arie di sapientoni, la verità è che nessuno mai è in grado di prevedere l'evoluzione; sì c'è qualcuno che ne intravvede delle possibilità, dei "futuri possibili" oserei dire, ma poi è solo questione di fortuna.

Secondo voi contano le nostre scelte o siamo trascinati dalla corrente del mercato?

Ancora grazie e buon lavoro!

Risponde Salvatore,

Bhe carissimo Antonio, a parte i complimenti a Jurassic News, confesso di averci capito pochino della tua e-mail. Comunque cercherò di mettere assieme qualche pensiero.

Penso anch'io che siamo trascinati dalla corrente del mercato e poco possiamo per far cambiare rotta al bastimento. Qualche possibilità esiste comunque, limitata e non tanto rivolta alla protesta sociale o contraria alla globalizzazione o agli OGM. Una presa di posizione personale, ecco tutto, che sia contraria al comune pensiero.

Forse dopo tre anni di Jurassic News ho assorbita una convinzione "etica", se mi passate il termine, una riflessione sul perché siamo così asserviti alle logiche di marketing e a quanto poco conosciamo degli oggetti che ci vengono propinati tutti i santi giorni, come fossero la panacea ai nostri mali.

Ad esempio la televisione. Ho sempre sentito e letto sui giornali una crescente insoddisfazione per i contenuti e il come questi ci vengono propinati, però nessuno si sogna di rinunciare alla televisione, perché?

In tempi di recessione e di "non si arriva alla fine del mese", ecco che i volantoni dei superstore sono

pieni di offerte per video LCD o plasma. Tutto a tasso zero, sta bene, ma poi queste rate bisogna pur pagarle... e si vendono "come il pane" mi ha confidato un amico che lavora presso uno di questi mega center tecnologici. Per non parlare di telefonini, una piaga sociale secondo me. E infatti lo tengo spesso spento, anche se è quello che mi ha fornito la ditta per la quale lavoro. Due volte al giorno lo accendo e spulcio la segreteria, lo trovo tanto più comodo che essere "always on-line", che orrore!

Prima dell'estate mi si è guastato il televisore, un "vecchio" Philips 100 Hz a tubo. Sapevo non valeva la pena ripararlo: l'ultima volta mi hanno chiesto il prezzo del nuovo per cambiarmi l'alimentatore! Ho consegnato TV e telecomando e pure il videoregistratore (anch'esso difettoso) al centro di raccolta differenziata (ci tengo a raccogliere in maniera differenziata i rifiuti, anzi invito caldamente tutti a farlo, se non per voi fatelo per i vostri figli) e... non ne ho comprati altri.

Panico di moglie e prole, ma più un disagio che una vera disperazione.

A distanza di mesi mia moglie è felice della scelta, i bambini si sono abituati e mai hanno chiesto di vedere qualche programma: hanno i loro DVD che vedono sul PC e questo basta. Invece abbiamo scoperto tutti, ma proprio tutti, che senza TV si può stare, che abbiamo più tempo per fare altre cose, che la radio è bellissima, soprattutto di sera e soprattutto Radio3.

Compriamo il giornale, ma non tutti i giorni e non sempre la stessa testata. credetemi, anche così siamo sommersi di informazione (chissà quanta di essa è veritiera...).

L'altroieri si parlava sul giornale di una polemica legata ad una certa pubblicità della TIM, credo, che in qualche modo avrebbe offeso non so che sensibilità femminile. Incredibile: non avevamo vista la pubblicità! Ci siamo riusciti, siamo oggi disintossicati e felici.

Ah, dimenticavo... ho mandato la raccomandata per disdire il canone RAI. Non mi illudo che smetteranno di inviarmi i bollettini e le ingiunzioni di pagamento, ma sai che gioia buttarli nel bidone giallo della carta da riciclare?

Jurassic News

Retrocomputer Magazine — Anno 3 - Numero 20 - Dicembre 2008

